

Estimación del tiempo de vida útil de las baterías de litio-ion, mediante la optimización de los hiper-parámetros del Kernel en un proceso Gaussiano con un algoritmo genético de valor real

Tesis para obtener el título de Magister en Investigación Operativa y Estadística

Estudiante: *Jaime David Parra Martínez*

Director: *Dr. Carlos Julio Zapata*



Universidad Tecnológica de Pereira

Facultad de Ingeniería Industrial

Maestría en Investigación Operativa y Estadística

Pereira, Risaralda

2017

AGRADECIMIENTOS

En primer lugar, quiero agradecerle a Dios por permitirme culminar de una manera exitosa esta parte de mi vida. A mi esposa y a mis padres les agradezco por su apoyo incondicional en cada una de los obstáculos que se me han presentado en esta etapa, mil gracias por permanecer a mi lado sin desfallecer nunca, ustedes saben lo mucho que he luchado por ello. A mi hermana y mi abuelita, gracias por sus palabras y constante motivación.

Quiero agradecerle también al Ph.D. Carlos Julio Zapata por las valiosas contribuciones realizadas al trabajo final, sinceramente valoro mucho el tiempo que ha dedicado para revisarlo, a pesar de tan diversas actividades que le ocupan.

Deseo reconocer también a cada uno de mis docentes que compartieron sus conocimientos con el único fin de enriquecerme intelectualmente. A mis compañeros de estudio, por darme la oportunidad de compartir con ustedes estos años, y vivir tantos momentos juntos.

Un agradecimiento especial al Ms.C Julián Gil González por compartir información valiosa para la culminación del documento final. Gracias por su amistad.

Finalmente, quiero dedicarle este trabajo de una manera muy sincera y especial a mi hija Antonella, pues es mi orgullo y mi motivación día a día, es quien me impulsa siempre a mejorar y ser un ejemplo de vida para ella. Espero algún día tengas la oportunidad de leer este trabajo.

A todas aquellas personas que de una u otra manera estuvieron presentes en mi evolución y en el desarrollo de mi trabajo de grado. Mil gracias. Los quiero.

RESUMEN

La estimación del tiempo de vida útil (RUL) debido a sus siglas en inglés "Remaining Useful Life", está definida como el espacio de tiempo en el cual un determinado elemento o componente seguirá cumpliendo de manera adecuada la labor para la cual fue diseñado, o contribuyendo de forma apropiada al sistema en el que se encuentra. Hoy en día, la estimación del RUL es una herramienta esencial en distintas áreas de la ciencia tanto teóricas como aplicadas, por ende ha incursionado en ramas como: bioestadística, econometría, mecánica, y en general, cualquier tarea en la que interviene la ingeniería. Esto último, se debe al hecho de que un adecuado cálculo el RUL permite tomar decisiones correctas en variables como confiabilidad, rendimiento y mantenimiento.

En la literatura, se han realizado diferentes enfoques para estimar el RUL de un dispositivo específico que se encuentra en estudio. Algunas de las metodologías que se han aplicado para resolver este tipo de problemas son: máquinas de soporte vectorial, modelos de riesgo, redes neuronales artificiales, y algunos otros modelos de tipo estocástico. Todos ellos como generalidad, usan metodologías basadas en gradiente para resolver el problema de estimación, lo cual implica que corren el riesgo de encontrar soluciones encontradas como mínimos locales en el espacio de soluciones, y no un mínimo global como sería lo esperado.

En el presente proyecto, se estima el tiempo de vida útil restante de las baterías de Litio ion que se encuentran en el repositorio de base de datos de la NASA. Para cumplir tal fin, se tiene en cuenta un proceso Gaussiano como metodología de regresión. Para que esta técnica opere de una manera correcta, es necesario definir una función de distribución previa o prior que asigne un valor promedio y una función de covarianza a los datos de entrada. Esta función de covarianza o Kernel se recomienda sea asumida de acuerdo al comportamiento que se espera tengan los datos de entrada (datos periódicos, datos constantes, etc). En este documento, se selecciona una función exponencial cuadrática para definir el Kernel, la cual se caracteriza por ser una función suave, que no genera cambios tan drásticos entre los datos de entrada (dinámica esperada por las baterías), y que fundamentalmente posee tres hiperparámetros (σ_f , λ y σ_n) que deben ser optimizados para el excelente rendimiento de la función en la estimación. Se llaman hiperparámetros ya que son parámetros definidos para la función de distribución prior o previa.

El proceso de optimización de estos hiperparámetros se efectúa mediante un algoritmo genético de valor, contando con la ventaja de que este tipo de herramientas desenvuelven

en todo el espacio de soluciones y su respuesta no está ligada fuertemente a un punto inicial, como lo hacen otro tipo de metodologías.

Al efectuar la estimación de estos hiperparámetros y al hacer las pruebas correspondientes, se concluye que para los datos de prueba de las baterías de Litio ion, se obtienen resultados que poseen menores errores en la predicción del tiempo de vida útil restante (RUL) de estos dispositivos, ya que son comparados con diversos documentos en la literatura que intentan de igual manera satisfacer o resolver este tipo de problema científico.

CONTENIDO

1. Antecedentes	11
2. Introducción	16
3. Objetivos	18
3.1 Objetivo general	18
3.2 Objetivos específicos	18
4. Marco teórico	19
4.1 Baterías Litio-ion	19
4.2 Procesos Gaussianos	21
4.3 Algoritmos genéticos	25
4.3.1 Codificación	27
4.3.2 Población	28
4.3.3 Función Objetivo	28
4.3.4 Selección	29
4.3.5 Recombinación	30
4.3.6 Mutación	33
5. Materiales y métodos	36
5.1 Descripción base de datos	36
5.2 Caracterización base de datos	39
5.3 Validación	44
5.4 Regresión para hallar el RUL	45
5.4.1 Optimización de los hiper-parámetros mediante algoritmo genético de valor real	48
6. Resultados	58

6.1 Filtración de ciclos	58
6.2 Regresión con proceso Gaussiano y Algoritmo genético de valor real.....	60
6.2.1 Selección parámetros del algoritmo genético	60
6.2.2 Aplicación del modelo propuesto	63
6.3 Comparación resultados Toolbox vs Metodología propuesta	66
6.4 Comparación resultados literatura	72
7. Conclusiones	76
8. Trabajo Futuro	77

LISTADO DE FIGURAS

Figura 1. Circuito eléctrico RC de primer orden para modelamiento batería litio-ion, Imagen extraída de [28]	20
Figura 2. Curvas $V_{OC} - SOC$ para baterías de Litio-ion [28]	21
Figura 3. Esquemático general algoritmo genético	26
Figura 4. Distintas codificaciones población algoritmo genético	27
Figura 5. Ejemplo método de selección por ruleta	30
Figura 6. Conceptualización recombinación heurística	32
Figura 7. Ejemplo numérico recombinación heurística.....	32
Figura 8. Conceptualización mutación no uniforme	34
Figura 9. Ejemplo numérico mutación no uniforme.	35
Figura 10. Estructura jerárquica de la base de datos "Battery" del Prognostics Data Repository de la NASA	38
Figura 11. Matriz Entrada/Salida para la estimación del tiempo de vida útil restante de las baterías de Litio-ion	43
Figura 12. División de la matriz de parámetros para el Training Set y Test Set	45
Figura 13. Población inicial para estimación de los hiperparámetros	48
Figura 14. Población de individuos con su respectiva función objetivo	49
Figura 15. Reemplazo de individuos en población del algoritmo genético de valor real ..	52
Figura 16. Diagrama de flujo proceso de estimación del RUL para baterías de Litio-ion	53
Figura 17. Diagrama de bloques general proceso de estimación del RUL de las baterías Litio-ion	56
Figura 18. Método de regresión para la batería 54 de la base de datos [26]	64
Figura 19. Ejecución del ToolBox de Matlab para la regresión de la batería 54 de la base de datos [26]	66

Figura 20. Rendimiento porcentual comparación metodologías	68
Figura 21. Comparación resultados metodología propuesta y ToolBox GPR para batería 5	69
Figura 22. Comparación resultados metodología propuesta frente al ToolBox GPR para batería 45	70
Figura 23. Comparación resultados metodología propuesta frente al ToolBox GPR para batería 41	71
Figura 24. Verificación RUL Estimado vs RUL Real para la batería 40	74
Figura 25. Verificación RUL Estimado vs RUL Real para la batería 47	74

LISTADO DE TABLAS

Tabla 1. Clasificación de ciclos para la base de datos del repositorio de la NASA para baterías de Litio-Ion	59
Tabla 2. Subdivisión ciclos de descarga baterías de Litio-Ion para datos de entrenamiento y prueba	60
Tabla 3. Valores hiperparámetros para tamaño población = 20 y con un máximo de 100 generaciones	61
Tabla 4. Valores hiperparámetros para tamaño población = 20 y con un máximo de 500 generaciones	61
Tabla 5. Valores hiperparámetros para tamaño población = 20 y con un máximo de 1000 generaciones	62
Tabla 6. Condensado de las desviaciones estándar de los hiperparámetros en las distintas corridas para la batería 54	63
Tabla 7. Valor hiperparámetros para cada procedimiento evaluado	65
Tabla 8. Resultados de comparación entre el ToolBox GPR Matlab y la metodología propuesta	67
Tabla 9. Vida útil baterías según información suministrada por [26]	73

LISTADO DE ECUACIONES

Ecuación 1. Representación general de un modelo de baterías litio ion	20
Ecuación 2. Tensión en los terminales de la batería Litio Ion en tiempo discreto.....	20
Ecuación 3. Tensión de equilibrio en modelo general de batería Litio Ion	20
Ecuación 4. Denotación general de un proceso Gaussiano	22
Ecuación 5. Modelo de suposición inicial para el proceso Gaussiano	22
Ecuación 6. Kernel exponencial cuadrático	23
Ecuación 7. Función de covarianza con ruido Gaussiano	23
Ecuación 8. Distribución conjunta de las salidas del modelo	24
Ecuación 9. Función media de la distribución posterior del modelo	24
Ecuación 10. Función de covarianza de la distribución posterior del modelo	24
Ecuación 11. Logaritmo de la verosimilitud marginal	24
Ecuación 12. Probabilidad i – ésimo individuo bajo selección por ruleta	29
Ecuación 13. Generación de descendientes bajo recombinación heurística	31
Ecuación 14. Mutación no uniforme para codificaciones de valor real.....	33
Ecuación 15. Operador de variación de distancia para la mutación no uniforme.....	33
Ecuación 16. Índice de fluctuación para el voltaje de descarga.....	40
Ecuación 17. Índice de fluctuación para la temperatura	40
Ecuación 18. Índice de asimetría para el voltaje de descarga	40
Ecuación 19. Índice de asimetría para la temperatura	41
Ecuación 20. Índice de curtosis para el voltaje de descarga	41
Ecuación 21. Índice de curtosis para la temperatura	41
Ecuación 22. Error medio cuadrático	65
Ecuación 23. Function score de comparación	72

1. ANTECEDENTES

En los últimos años, tanto la industria como la academia vienen presentando un gran interés en desarrollar técnicas y modelos que permitan conocer en qué estado operativo se encuentra un determinado dispositivo o componente, todo esto con el fin de evaluar la cantidad de tiempo para el cual dicho elemento seguirá siendo productivo y útil en el ambiente que se encuentre implementado [1].

Profundizando un poco, lo anterior denota que el pronóstico del tiempo de vida útil restante (RUL) es un concepto esencial para la industria actual a nivel mundial, puesto que el hecho de tener maquinaria segura, confiable y con un alto nivel de disponibilidad es considerado como un ítem de primera importancia, por lo cual se ha venido aplicando de manera exitosa en diversos dominios de la ciencia.

Por ejemplo, en la industria aeroespacial [2,3], usan el RUL para evaluar la confiabilidad de las estructuras mediante métodos probabilísticos o en algunos casos procesos determinísticos, si se cuenta con la cantidad de datos suficientes como: pruebas hechas a los distintos materiales y estructuras de los aviones. El RUL también ha incursionado en el área de la ingeniería electrónica [4], donde se efectúa un pronóstico de tiempo útil de operación a todos los elementos electrónicos (tarjetas seriales y simuladores de satélite) usados en el desarrollo del sistema de posicionamiento espacial; Allí, los parámetros de caracterización de cada módulo, se extraen de estudios previos donde se correlacionan ciertos parámetros físicos relevantes y la condición de deterioro o degradación del dispositivo. Además de las ramas anteriores, se destaca una aplicación del RUL en el ámbito de la ingeniería nuclear [5,6] donde se estima la confiabilidad, seguridad y sostenibilidad de todos los componentes que conforman las fuentes de energía de plantas nucleares, con el fin de evitar mantenimientos frecuentes que representaban un costo alto para las compañías. Se resalta además, que la estimación del tiempo de vida útil restante ha sido aplicado también a ciencias como la medicina [7] y la ejecución de pronósticos del clima [8].

En la industria Colombiana por ejemplo, se está practicando una nueva metodología que pretende determinar el tiempo útil remanente de los transformadores de distribución de 0 a 10 KVA, con la ayuda de un dispositivo de microondas que analiza las ondas magnéticas transmitidas por estos mientras se encuentran en funcionamiento; todo con el fin de programar de manera más eficaz y temprana mantenimientos preventivos y correctivos, lo cual ahorraría considerables sumas de dinero, debido al hecho de tener información sobre el estado actual del transformador, sin necesidad de realizar algún corte de energía [57].

Ahora bien, diversos enfoques se han utilizado en la literatura con el fin de dar la mejor solución a este crítico tema de ingeniería; estas diferentes tendencias, se pueden subdividir en dos grandes grupos, los cuales son: métodos basados en técnicas probabilísticas y técnicas no probabilísticas [9].

Distintos casos de éxito se han dado bajo la estimación del RUL mediante técnicas no probabilísticas. Por ejemplo, las redes neuronales han sido ampliamente usadas para este tipo de problemas de deterioramiento de dispositivos, ya que una de sus características principales es tener la capacidad de interpretar la información o experiencia inicial y generar un aprendizaje adaptativo para ejecutar nuevas tareas, sin necesidad de elaborar modelos a priori ni funciones de distribución de probabilidad. En [10,11] se efectúan predicciones del tiempo de vida remanente de dos dispositivos diferentes: baterías y rodamientos respectivamente. Allí, se crean distintos modelos de entrenamiento para la red, y a su vez, distintas salidas que permiten generar respuestas a la necesidad de conocimiento de vida útil inicial para dichos elementos.

Otro tipo de modelos no probabilísticos que han incursionado en la resolución del RUL son aquellos que se denominan: "basados en la experiencia". [12] muestra un claro prototipo de este tipo de técnicas, puesto que expone que la estimación del tiempo de vida útil restante puede resolverse mediante la recopilación de ejemplos que muestren distintos casos de degradación de componentes similares, y posteriormente se compara una nueva entrada con el historial previo existente, para así dar solución al valor del RUL necesitado. Este tipo de métodos necesitan la extracción de una serie de indicadores de bienestar que permitan conocer en qué estado se encuentra el dispositivo en estudio (caracterización o parametrización); dichas variables se construyen mediante modelos de regresión lineal múltiple, la cual es otra técnica no probabilística usada para el cálculo del RUL. [13] representa un ejemplo de esta última metodología, ya que allí desarrollan un procedimiento que permite extraer indicadores de operación para rodamientos de motores, teniendo en cuenta como principal entrada los datos de vibración que se tenían.

Por otro lado, se encuentran los métodos probabilísticos, los cuales generan una salida del RUL en términos de probabilidad [9]. Para ello, generalmente se modela el fenómeno de deterioramiento de los dispositivos como una variable aleatoria, lo cual los induce a usar herramientas estocásticas para poder trabajar con su comportamiento. Por ejemplo, es lógico pensar que el deterioro de un sistema o dispositivo vaya aumentando a través del tiempo debido a diversos factores climáticos, estructurales y demás, por lo cual un modelamiento estocástico basada en un proceso Gamma se hace indicado para representar lo anterior, ya que se conoce que este es un proceso estocástico monótonamente creciente [14]. En la referencia [15] se muestra como un proceso

Gamma es aplicado a sistemas de estimación de vida útil por medio de cálculos de confiabilidad. Existe otra técnica o método probabilístico muy usado en la literatura para calcular el tiempo de vida útil restante (RUL), este es el proceso Wiener, cuyo fundamento esencial es el hecho de que en algunos casos los sistemas en general no presentan comportamientos de deterioro monótonamente crecientes como lo propone un proceso Gamma [16], sino que pueden existir variaciones en su cantidad de uso y en la frecuencia de los mantenimientos preventivos, haciendo que la máquina o sistema tenga trayectorias de deterioro variables en el tiempo [17].

Este último método probabilístico, fue usado en [18] para mostrar de manera práctica los resultados obtenidos para la estimación del tiempo de vida útil restante de unos giroscopios en un sistema de navegación inercial.

Una de las herramientas más relevantes usadas en las diversas investigaciones para resolver la estimación del tiempo de vida útil remanente es el proceso Gaussiano, el cual es una herramienta transversal usado en diversos campos de la ciencia y forma parte del grupo de modelos probabilísticos. Este instrumento es en definición, una colección de cualquier número finito de variables aleatorias, las cuales poseen o tienen una distribución Gaussiana conjunta, donde dependiendo de la característica de la salida, este tipo de herramienta puede ser útil tanto para regresión si se desea que la salida sea continua, o de clasificación cuando se tiene la necesidad que el resultado sea discreto [19]. Considerando lo anterior y para el caso de la estimación del RUL, el proceso Gaussiano es utilizado para solucionar un problema de regresión (GPR), cuya salida es un intervalo de confianza en el cual se espera que esté presente la vida útil remanente del dispositivo que se esté analizando, o por lo menos una variable que permita calcularlo. Es esencial resaltar, que este intervalo de confianza obtenido, depende en gran medida de unos parámetros libres que posee el proceso Gaussiano en su función Kernel o de covarianza conocidos como *híper-parámetros*, y su optimización es en efecto fundamental para obtener excelentes resultados en el proceso de regresión. Su nombre se debe a que son parámetros que deben ser estimados mientras se está encontrando un modelo que se ajuste a los datos de entrada (base de datos), es decir, son parámetros que se estiman mientras se asume una función de distribución prior o previa para el proceso.

Como se ha podido observar, diversos tipos de técnicas se han utilizado a nivel mundial a través del tiempo para calcular el tiempo de vida útil restante de diversos dispositivos (RUL). Ahora bien, con el objetivo de ahondar más en la esencia de este documento, se habla de distintos ejercicios donde se han aplicado los GPR a nivel mundial para lograr este tipo de estimación a los dispositivos. Por ejemplo, en [20], mediante una serie de datos recolectados a partir del muestreo de ocho acelerómetros, se calcula el tiempo de

vida útil de unos rodamientos con la ayuda de un proceso Gaussiano. Allí, los hiper-parámetros de la función de covarianza se calculan mediante un optimizador basado en gradiente. Aunque en [21] también se usa un proceso Gaussiano para estimar el RUL de unos rodamientos, en este documento se expone una integración de varios modelos GPR (Procesos Gaussianos de regresión) con el fin de componer funciones de covarianza compuestas que tengan un mejor rendimiento en la predicción. Aquí los hiper-parámetros se hallan mediante inferencia dados los datos de entrenamiento.

Ahora bien, a nivel local hay gran variedad de temas que se han tratado con ayuda de este tema, debido a su gran flexibilidad en su estructura matemática. Por ejemplo, dos colaboradores en la universidad del Quindío, proponen en [53] un modelo en el que intervienen los procesos Gaussianos de regresión para valorar y predecir en un intervalo de tiempo el nivel de cicatrización de nueve conejos, a los cuales se les efectuó una resección de piel. Además de lo anterior, en [54] se desarrolla un método similar, pero esta vez con el objetivo de predecir el aporte de los componentes espectrales de unas imágenes tomadas a los frutos de banano para el siguiente día de maduración.

Particularmente en la Universidad Tecnológica de Pereira también se ha trabajado en el ámbito de los GPR, los artículos [55] y [56] muestran dos temas aislados tratados bajo el mismo concepto. [55] expone como detectar eventos sonoros en señales de música usando la transformada de Fourier y clasificadores binarios basados en procesos Gaussianos. Por su parte [56], trata el tema de expansión de telefonía móvil (red GSM) mediante la caracterización de las celdas operativas a nivel nacional con variables como: número de conexiones caídas, número de conexiones rechazadas, manejo del recurso, entre otras.

Aunque el proceso Gaussiano es una herramienta transversal que está siendo usada en varios campos de la ciencia, y se viene aplicando con un gran auge en todos los temas relacionados con el aprendizaje de máquina, éste presenta una desventaja latente, y se debe al hecho de la forma en que resuelve el problema de estimación de los hiper-parámetros, puesto que los métodos de optimización usados para el cálculo de ellos se centran en técnicas basadas en gradientes descendentes [22,23] o gradientes conjugados [24], lo cual conlleva a que los valores encontrados sean mínimos locales y no mínimos globales como es lo esperado.

Lo anterior, se debe principalmente al hecho de que los métodos basados en gradiente operan de una buena manera si la función que se trata de estimar que relacione los datos de entrada y salida es convexa, además este tipo de técnicas son muy susceptibles al punto inicial, ya que a partir de este punto, se empieza a avanzar hacia la minimización de la función objetivo, y una mala selección de este incurriría en resultados muy pobres.

Ahora bien, desde el año 2008 donde fue realizada la conferencia "IEEE Prognostics and Health Management" [9,10], el tema del cálculo del RUL para las baterías de litio ion se ha vuelto crítico en el ámbito mundial, puesto que juegan un papel fundamental en la autonomía de cualquier dispositivo eléctrico y electrónico, como por ejemplo actualmente lo son los vehículos eléctricos [25]. Es por lo anterior, que en este trabajo se propone la estimación del tiempo de vida útil restante o remanente de las baterías de Litio Ion a partir de un proceso Gaussiano de regresión, cuyos hiperparámetros sean calculados a partir de la minimización de una función de verosimilitud, optimizada usando un algoritmo genético de valor real. Este tipo de procedimiento va a permitir comparar el rendimiento puntual de la optimización basada en gradientes con uno centrado en algoritmos evolutivos. La base de datos de baterías de litio ion usada es suministrada por la NASA en su repositorio virtual [26].

2. INTRODUCCIÓN

Alrededor del año 2008, donde se tuvo la conferencia dictada por la IEEE denominada "Prognostics and Health Management", el auge por la estimación del tiempo de vida útil remanente o RUL por sus siglas en inglés ha tenido un crecimiento importante con respecto a su interés por parte de los entes científicos [51], ya que se ha convertido en un tema crítico para cualquier ámbito bien sea investigativo o empresarial tomando como consideración el diseño de maquinaria. En esta conferencia, se recalca esencialmente la importancia de poder predecir fallas incipientes en la maquinaria, para así maximizar retornos de inversión y optimizar los cronogramas de mantenimiento periódico.

Cada vez más la industria y en general todo lo perteneciente al desarrollo tecnológico, se encuentran ingresando en una etapa de despliegue hacia dispositivos que le permitan a la maquinaria ser más autónoma, elementos que posean la bondad de brindar una mayor independencia al sistema en general, este es el caso puntual de las baterías de Litio-ion.

Considerando el ingreso a la era electrónica, recientemente las baterías de Litio-ion han sido usadas de manera frecuente para distintas aplicaciones como: vehículos eléctricos, sistemas de celdas fotovoltaicas, dispositivos electrónicos móviles, etc. Este uso ya común de las baterías de Litio-ion, se debe principalmente a su virtud química de tener una alta densidad de energía, de ser más livianas que otro tipo de baterías y principalmente por su característica de no poseer memoria [27]. Debido al apogeo de este tema y a la utilidad frecuente que se le está dando actualmente al dispositivo, se decide en el presente documento hacer la estimación del tiempo de vida útil remanente de las baterías de Litio-ion, tomando como referencia la base de datos suministrada por la NASA en su repositorio virtual [26].

Diversos enfoques se han realizado en los últimos años con el fin de dar solución a este caso de estudio, por ejemplo, la metodología usada en [12] propone estimar el RUL tomando como base la construcción de una base de datos en el cual se tenga ya el tiempo de vida útil remanente de otros dispositivos y calcular por ende el de las baterías de Litio-ion por similitud con estas. En [28] se revisan diversos aspectos de estos dispositivos, y condensan una serie de técnicas, modelos y algoritmos para el cálculo del RUL de ellos. Uno de los enfoques más recientes a este problema ha sido el expuesto en [58], en donde se extrae un indicador de los datos que se tienen y se efectúa una comparación de la capacidad real de la batería y el parámetro extraído mediante un modelo Gray optimizado.

En paralelo a la necesidad de la estimación del RUL, también el aprendizaje de máquina ha tenido mucho crecimiento a nivel de aplicación en el ambiente académico, es por lo

anterior que hay algunos autores también han dado su aporte en la estimación del tiempo de vida útil remanente de las baterías de Litio-ion. [46,49,59] son un claro ejemplo de lo anterior, ya que cada uno de ellos propone una metodología diferente para la resolución del problema: máquinas de soporte vectorial, procesos Gaussianos y redes neuronales respectivamente.

La mayoría de este tipo de métodos probabilísticos aplicados tienen dificultad a nivel de implementación física, debido a que trabajan teóricamente con parámetros internos de la batería que son extraídos en laboratorios con dispositivos diseñados para tal fin. Por lo anterior, el presente proyecto propone utilizar un proceso Gaussiano de regresión teniendo como parámetros de entrada variables que puedan ser de fácil acceso para el ejecutante como lo son tensión de la batería y temperatura de la misma, con el objetivo principal de poder aplicar el modelo resultante en trabajos a nivel industrial. Se pretende finalmente predecir el tiempo de vida útil remanente de una batería de Litio-ion mediante la estimación de la disminución porcentual de la capacidad nominal del dispositivo dada en amperes-hora (Ah).

Para el buen funcionamiento del proceso Gaussiano de regresión se deben calcular tres hiperparámetros (σ_f , λ y σ_n) que son calculados en la literatura mediante métodos basados en gradiente [20,21,22,23,24,48,49,50]; para el presente documento, estos hiperparámetros se estimarán mediante un algoritmo genético de valor real, el cual es una herramienta que permite buscar en todo el espacio de solución establecido y no depender en gran medida del establecimiento de un punto inicial como lo hacen los métodos basados en gradiente.

La teoría perteneciente a cada uno de las herramientas que componen la metodología usada para el cálculo de RUL de las baterías de Litio-ion se presenta en la sección 4. A su vez, en la sección 5 se presenta toda la descripción del modelo, su etapa de caracterización, contenido y validación. Finalmente, en la sección 6 se obtienen los resultados obtenidos para la base de datos [26], y se compara la metodología propuesta con otras de éxito en la literatura. En la sección 7 y 8 se pueden encontrar las conclusiones y el trabajo futuro respectivamente.

3. OBJETIVOS

3.1 OBJETIVO GENERAL

Desarrollar una metodología de regresión basada en procesos Gaussianos que permita estimar el tiempo de vida útil restante (RUL) para baterías de Litio-ion usando como información sus ciclos de carga/descarga, siendo estimados los hiperparámetros del Kernel mediante un algoritmo genético.

3.2 OBJETIVOS ESPECÍFICOS

1. Caracterización de la base de datos de las baterías de Litio-ion, por medio de la extracción de patrones relevantes a las series de tiempo de los ciclos de carga y descarga de estas.
2. Desarrollar una metodología que permita estimar los hiperparámetros del Kernel del proceso Gaussiano, mediante un algoritmo genético de valor real.
3. Comparar el algoritmo diseñado con esquemas implementados y exitosos, para evaluar su rendimiento.

4. MARCO TEÓRICO

4.1 BATERIAS LITIO - ION

La batería de Li-ion o con su nombre técnico completo "batería de iones de litio", es básicamente un dispositivo creado con el fin exclusivo de generar un mejor almacenamiento de la energía eléctrica por medio de una sal de litio usada como electrolito. Este, permite producir una reacción electroquímica entre el ánodo y el cátodo generando finalmente la corriente eléctrica necesaria para excitar cualquier sistema. Para efectuar este proceso, una celda de litio-ion debe estar compuesta de cinco regiones: Un electrodo negativo colector de corriente, un electrodo de inserción negativo con compuesto poroso, un separador de poros, un electrodo de inserción positiva con compuesto poroso y finalmente un electrodo positivo colector de corriente . Su sistema de almacenamiento de energía puede estar compuesto de numerosas celdas de carga conectadas bien sea en configuración serie y/o en paralelo para cumplir bien sea especificaciones de voltaje y potencia [27].

Varios modelos se han expuesto en la literatura para representar las baterías de Litio ion, entre ellos uno de lo más usado es el modelo de tiempo real, en el cual se intenta garantizar veracidad sin sacrificar complejidad, por medio del uso de circuitos eléctricos que permiten exponer y caracterizar de una manera fiel la dinámica de voltaje-corriente de las celdas [28]. Lo anterior indica, que este modelo genera una razonable descripción de la dinámica de una batería de Litio-ion durante sus ciclos de carga/descarga [27].

En la figura 1, se puede verificar el modelo de tiempo real expuesto en [28], para el cual se tiene en cuenta una diferencia de potencial $V_h(k)$, considerada como el efecto histéresis entre las curvas de carga y descarga de las baterías de litio [29]. En este modelamiento, $V_h(k)$ hace parte del cálculo del voltaje de circuito abierto V_{OC} , junto con la tensión promedio de equilibrio de la $V_s(SOC)$, que depende del estado de carga que tenga la batería en un determinado instante de tiempo.

A su vez, se tienen en cuenta una serie de resistencias y capacitores que juegan un papel fundamental en el desarrollo del modelo.

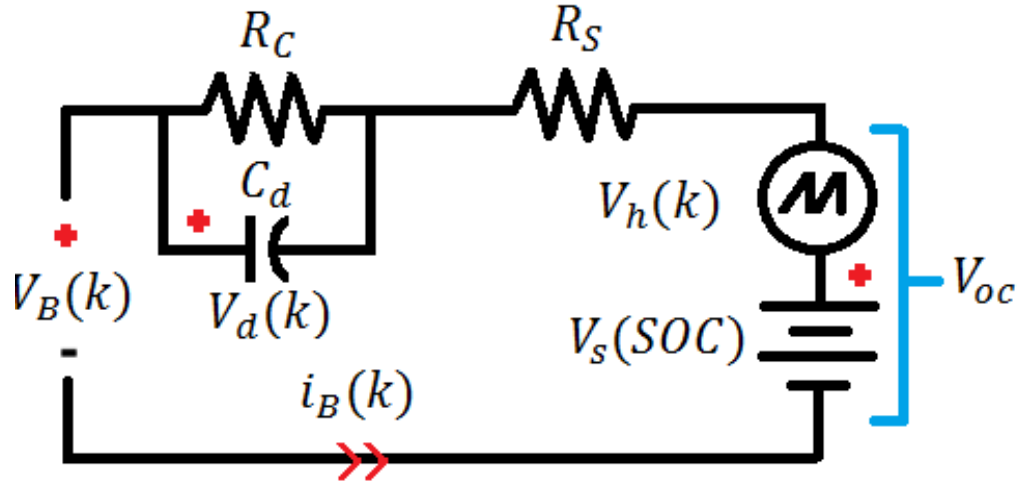


Figura 1. Circuito eléctrico RC de primer orden para modelamiento batería litio-ion.
Imagen extraída de [28].

Por ejemplo, la resistencia en serie R_S es el elemento encargado de suministrar información de la energía perdida por la batería en sus respectivos ciclos de carga y descarga. A su vez, R_C y C_d representan la etapa de transferencia de carga y finalmente $V_B(k)$ presenta la diferencia de potencial final en los terminales de la celda. Un modelo eléctrico para el modelo de la batería en tiempo discreto, es el siguiente [28]:

$$X(k+1) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \gamma & 0 \\ 0 & 0 & H \end{bmatrix} X(k) + \begin{bmatrix} \frac{-\eta T_s}{C_{max}} & 0 \\ R_C(1-\gamma) & 0 \\ 0 & (H-1)\text{sign}(i_B) \end{bmatrix} \begin{bmatrix} i_B(k) \\ V_{hmax} \end{bmatrix} \quad (1)$$

$$y(k) = V_B(k) = V_{OC}(SOC(k)) - V_d(k) - R_S i_B(k) + V_h(k) \quad (2)$$

$$V_S(SOC) = a_0 \exp(-a_1 SOC) + a_2 + a_3 SOC - a_4 SOC^2 + a_5 SOC^3 \quad (3)$$

Donde, $X(k+1) = [SOC(k+1) \ V_d(k+1) \ V_h(k+1)]^T$ representa el estado, $y(k)$ muestra la medida de salida de la batería, k simboliza la indexación en el tiempo, C_{max} expone la capacidad máxima de la batería y T_s indica el periodo de muestreo. Además se tiene un circuito RC alterno del cual se tiene que: $\gamma = \exp\left(\frac{-T_s}{\tau}\right)$, donde $\tau = R_C i_d$, $H(i_B) = \exp(-\rho * |i_B| * T_s)$.

Para condensar, las ecuaciones (1), (2) y (3) representan el voltaje de salida en los bornes de la batería de Litio ion considerando todas sus variables internas, como: caídas de tensión, resistencias internas totales y su voltaje de circuito abierto. El entendimiento de las variables que intervienen en esta representación permite conocer la dinámica en el funcionamiento de las baterías de Litio-ion en función de su tensión o estado de carga $V_S(SOC)$.

Finalmente, los coeficientes a_j de la ecuación (3) con $0 \leq j \leq 5$ son los coeficientes usados para realizar la parametrización de la curva $V_{OC} - SOC$ mostrada en la figura 2, la cual me indica cómo va cambiando la tensión en los bornes de la batería de acuerdo a su porcentaje de estado de carga.

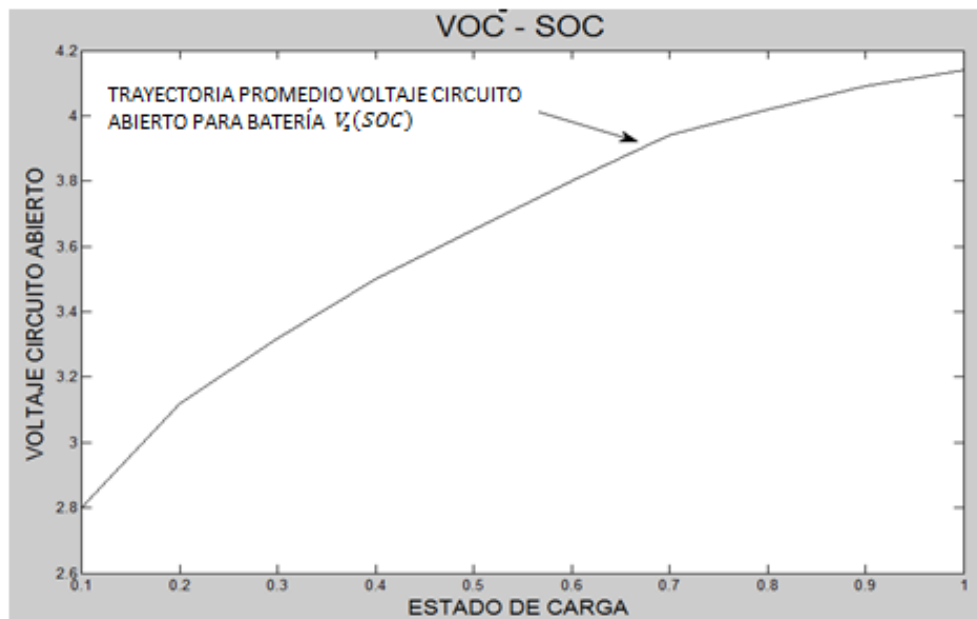


Figura 2. Curvas $V_{OC} - SOC$ para baterías de Litio-ion [28].

4.2 PROCESOS GAUSSIANOS

En un problema típico de aprendizaje de máquina supervisado, se tiene acceso a un conjunto de datos de entrenamiento $D = \{X_i, Y_i\}_{i=1}^N$, donde X_i se conoce como el vector de características y Y_i se denominan las salidas. Dependiendo de la naturaleza de las salidas Y , se reconocen dos tipos de problemas supervisados, si Y_i pertenece a los números enteros se denomina clasificación, por su parte, si Y_i pertenece a los reales se habla de un problema de regresión.

Para el caso de la estimación del tiempo de vida útil restante RUL, las características corresponden a parámetros extraídos del comportamiento de las baterías bajo diferentes circunstancias o condiciones, y su salida Y_i corresponde a su capacidad la cual se expresa en Amperes-hora (Ah). Claramente, la salida de este problema en particular toma valores pertenecientes a los números reales, de esta manera, la estimación del RUL configura un problema de regresión.

En este trabajo, dicha regresión se llevará a cabo a partir de un modelo probabilístico basados en procesos Gaussianos, el cual será especificado a continuación.

Como definición básica, un proceso Gaussiano es una colección finita de variables aleatorias, las cuales tienen una distribución conjunta Gaussiana. Este tipo de proceso estocástico está totalmente especificado por dos funciones: La función de media $m(x) = E[f(x)]$ y la función de covarianza $k(x, x') = E[(f(x) - m(x))(f(x') - m(x'))]$.

De esta manera, un proceso Gaussiano se denota como:

$$f(x) \sim GP(m(x), k(x, x')) \quad (4)$$

En el caso de regresión, las variables aleatorias están representadas por $f(x)$, la cual hace alusión a la capacidad de las baterías de Litio-ion, y la cual es función de los parámetros representados en el vector X_i .

En el formalismo Bayesiano, se hace necesario especificar como primer elemento un modelo inicial o prior para proceder a realizar el modelo de regresión. En esta etapa, es donde se manifiesta la creencia inicial de cómo se comportan los datos antes de observar o analizar los datos que se tienen [19]. Por lo anterior, para el presente trabajo se asume un proceso Gaussiano como modelo prior sobre las variables aleatorias f , el cual captura la suposición inicial del modelo para el comportamiento de las variables aleatorias que se encuentran consignadas en la base de datos [26].

Dicho modelo Gaussiano inicial o prior, se selecciona con una función media igual a cero y su función de covarianza $K(X, X)$ es calculada a partir de la función Kernel exponencial cuadrática. Estas expresiones se pueden verificar en las ecuaciones (5) y (6) respectivamente.

$$f \sim N(0, K(X, X)) \quad (5)$$

$$K(X_i, X_j) = \sigma_f^2 e^{\left(-\frac{|X_i - X_j|^2}{2\lambda^2}\right)}. \quad (6)$$

Como se puede observar en la ecuación (6), esta función Kernel cuenta con dos hiperparámetros definidos como λ y σ_f . El primero de ellos (λ) representa la longitud de escala del proceso y describe que tan suave es la función, es decir, longitudes de escala pequeños significa que los valores pueden cambiar rápidamente, mientras que un λ alto caracteriza a funciones que cambian muy lentamente. Este parámetro además, me representa que tan lejos de manera confiable puedo extrapolar en los datos de entrenamiento X .

Por otro parte, σ_f es básicamente un factor de escala que se le aplica a cualquier función de covarianza e indica la variación de los valores de la función respecto a su media. Valores muy altos de σ_f permite un mayor variación, mientras que σ_f pequeños caracteriza a funciones que permanecen cerca a su media.

Cabe anotar, que este tipo de función Kernel se selecciona para el presente trabajo, ya que se espera que ciclos que se encuentren cerca en el espacio de entrada están muy correlacionados, y que dicha relación va disminuyendo a medida que se van alejando dichos datos, es decir, el Kernel exponencial cuadrático se usa para modelar funciones que no posean cambios muy drásticos en su dinámica, como es el caso de las baterías de Litio-ion, puesto que es válido considerar que ciclos contiguos de descarga en la batería no difieran mucho y estén altamente correlacionados.

Ahora bien, la idea de la regresión no es solo generar un modelo que se adapte a los datos de entrenamiento, sino que también pueda predecir de manera eficiente el valor de la variable aleatoria ante nuevos datos X^* . Sin embargo, en modelos realistas, para el entrenamiento no se tiene acceso al valor de la función aleatoria f sino a una versión ruidosa de esta, tal que $y = f(x) + \varepsilon$, es decir, que el proceso Gaussiano permite la existencia de una señal aleatoria aditiva conocida como ruido (ε) para los datos de entrenamiento X en la estimación del RUL para las baterías de Litio ion.

Se asume que este ruido sigue una distribución Gaussiana idénticamente distribuida con varianza σ_n^2 . Este hiper-parámetro representa cuanto ruido se espera que exista en los datos que ingresará al modelo. Con lo anterior, la covarianza sobre las nuevas observaciones ruidosas se convierte en lo siguiente:

$$cov(y) = K(X, X) + \sigma_n^2 I \quad (7)$$

En este sentido, la distribución conjunta de las salidas de entrenamiento f y las salidas del conjunto de prueba f^* de acuerdo al prior definido está dado como [19] :

$$\begin{bmatrix} y \\ f^* \end{bmatrix} \sim N \left(0, \begin{bmatrix} K(X, X) + \sigma_n^2 I & K(X, X^*) \\ K(X^*, X) & K(X^*, X^*) \end{bmatrix} \right) \quad (8)$$

A partir de la distribución conjunta definida, es posible muestrear variables latentes como función de los datos de entrenamiento X y los datos de prueba X^* , sin embargo, las únicas funciones que se pueden tener en cuenta son aquellas que se ajustan a los datos. El proceso anterior, se lleva a cabo a partir del cálculo de la distribución Posterior, la cual se puede determinar condicionando la distribución Gaussiana conjunta sobre las observaciones. Esta distribución Posterior puede determinarse de manera analítica o exacta siempre y cuando el Prior asumido siga una distribución Gaussiana, para este caso, como se cumple lo anterior, el Posterior está dado por las siguientes expresiones:

$$f^*|X, y, X^* \sim N(\bar{f}^*, cov(f^*)) ,$$

donde

$$\bar{f}^* \triangleq E[X, y, X^*] = K(X^*, X)[K(X, X) + \sigma_n^2 I]^{-1}y. \quad (9)$$

$$cov(f^*) = K(X^*, X^*) - K(X^*, X)[K(X, X) + \sigma_n^2 I]^{-1}K(X, X^*). \quad (10)$$

Básicamente, las ecuaciones (9) y (10) son la forma para calcular la media y la covarianza del proceso Gaussiano asociado con la predicción de la capacidad remanente para la batería caracterizada por el vector X^* .

Analizando todo lo descrito anteriormente, se puede observar que el Kernel usado para el proyecto, tiene una serie de parámetros libres que deben de ser estimados. Estos parámetros se determinan a partir de la minimización del negativo del logaritmo de la verosimilitud marginal, también conocida como “La evidencia”, la cual está dada por la siguiente expresión [19]:

$$\log p(y|X) = -\frac{1}{2}y^T(K(X, X) + \sigma_n^2 I)^{-1}y - \frac{1}{2}\log|K(X, X) + \sigma_n^2 I| - \frac{n}{2}\log 2\pi. \quad (11)$$

La optimización de esta función de verosimilitud marginal para encontrar los hiperparámetros óptimos $(\sigma_f, \lambda, \sigma_n)$ es comúnmente llevada a cabo a partir de métodos

basados en gradiente, sin embargo, estos son susceptibles a la inicialización, puesto que una indebida elección de este podría conllevar a encontrar mínimos locales, por lo cual en el presente trabajo la estimación de dichos parámetros será llevada a cabo haciendo uso de un algoritmo genético de valor real, el cual opera sin restricción por todo el espacio de soluciones.

4.3 ALGORITMOS GENÉTICOS

El algoritmo genético es una técnica de búsqueda adaptativa la cual se deriva de la teoría Darwiniana de selección natural [30], esta expone que individuos o seres que posean unas mejores características dentro de una población determinada, tendrán mejores oportunidades de poder quedarse en ella y así de esta manera, excelentes chances de sobrevivir. Lo anterior denota, que los algoritmos genéticos son en esencia una analogía directa con la conducta natural, en donde se intenta descubrir individuos (soluciones) con mayor éxito de supervivencia, debido a la combinación de buenas características provenientes de sus antecesores o ancestros.

Este tipo de algoritmos son muy usados en la literatura para tratar diversos tipos de inconvenientes [31,32], puesto que es una herramienta robusta que es transversal en cualquier campo de la ciencia que exista la optimización, dando solución con éxito a dificultades que otro tipo de métodos encuentran diversos problemas.

El mecanismo de operación de un algoritmo genético como se expuso en párrafos anteriores, se centra en recrear la analogía que posee este con la selección natural, es decir: se cruzan dos individuos de la población con unas determinadas características, esta relación va a producir un nuevo individuo descendiente de sus padres, cuyos parámetros finales tendrán que estar relacionados con los de sus predecesores. Este nuevo organismo, es evaluado para determinar que tan útil es como solución, e implícitamente se determinará si vale la pena que sus características se vayan propagando a través del tiempo para encontrar una excelente solución [33].

Un algoritmo genético posee varios subprocesos que deben ser tenidos en cuenta y llevados a cabo, para desarrollar un proceso robusto que lleve finalmente a obtener soluciones exitosas. Entre los procesos más relevantes, se encuentran los siguientes: [33]:

- Codificación
- Población
- Función Objetivo
- Selección

- Recombinación
- Mutación

Un diagrama de flujo básico que representa el esquema general de un algoritmo genético, se presenta en la figura 3.

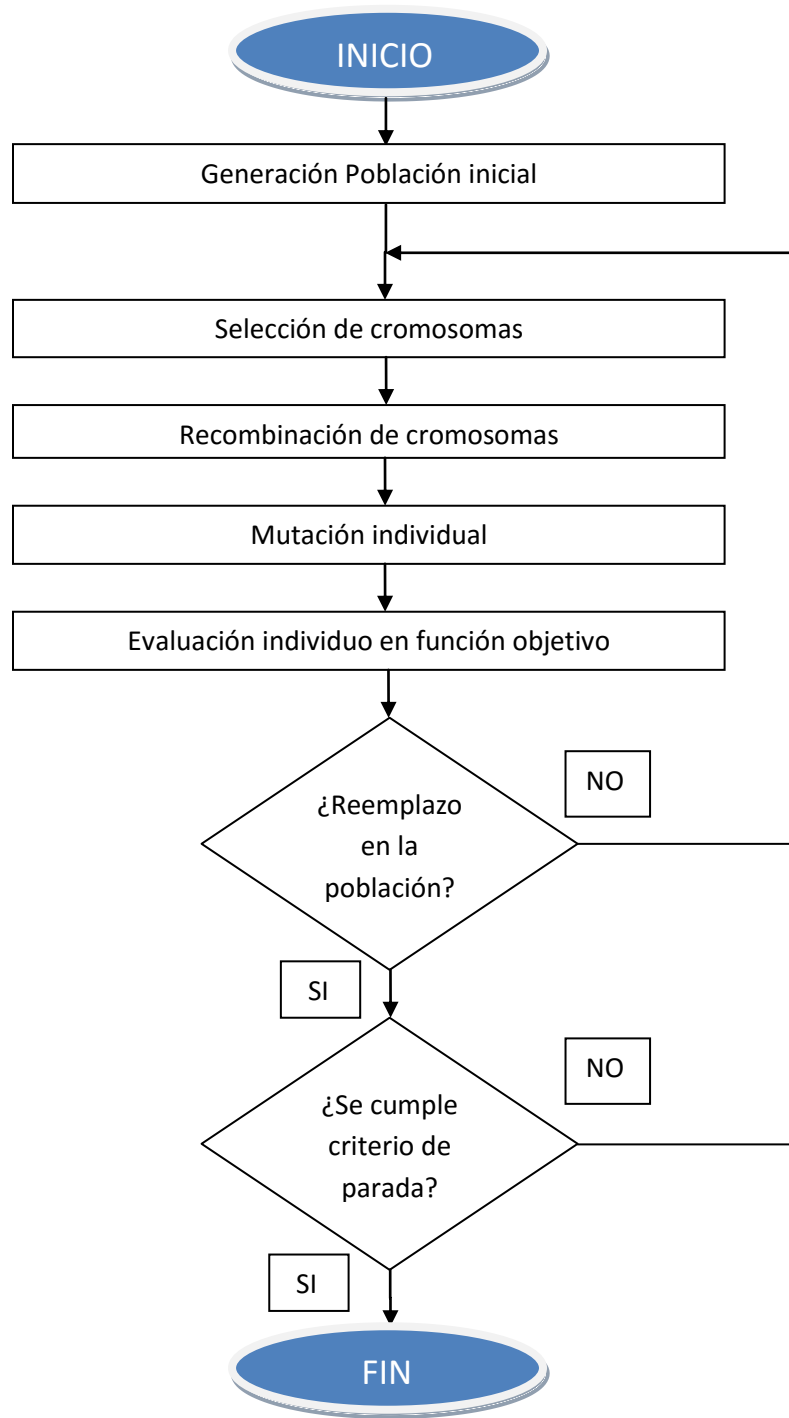


Figura 3. Esquemático general algoritmo genético

Enseguida, se expone en qué consisten cada uno de los subprocesos que conforman el algoritmo genético.

4.3.1 CODIFICACIÓN

Es un término usado para referirse a la manera de presentación de los datos de un algoritmo genético. La codificación representa la manera en la cual se muestra cada uno de los individuos en la población total. Es muy importante resaltar, que la codificación es esencial al momento de entender y operar un algoritmo genético, puesto que la función objetivo debe tener claro como evaluar dicho organismo.

En [34] por ejemplo, usan una codificación binaria para los datos que se manejan, representando los cromosomas como una cadena de caracteres que contiene 1's y 0's. Existe también otro tipo de representación, conocida como codificación de valor real [33], en la cual se tratan los individuos de la población como un conjunto de valores reales. Un ejemplo de lo anterior, se puede verificar en la figura 4.

Individuo		Individuo	
i	1 0 0 0 1 1 0 0 1 1 1 1 0 1	i	-23.2 -13.25 18.2
i + 1	0 1 0 0 1 1 0 0 1 0 1 0 1 0	i + 1	14.7 22.8 -1.2
Individuos codificados de forma binaria		Individuos codificados con valores reales	

Figura 4. Distintas codificaciones población algoritmo genético.

Considerando que el cálculo de las funciones objetivos se realiza para determinar si un nuevo organismo o individuo es bueno, y este cómputo se efectúa usando únicamente valores reales, se hace innecesario tener un sistema que condense la información de los individuos de manera binaria, puesto que la operabilidad de dichos datos requiere un tipo de procesamiento adicional de codificación y decodificación [33]. Por lo anterior, el presente proyecto trabaja con algoritmos genéticos de valor real, que permiten tener una mayor estabilidad mientras se efectúa las operaciones matemáticas, con la ventaja principal que el rendimiento computacional generado va a ser menor, debido al ahorro de ciertos cálculos de conversión.

4.3.2 POBLACIÓN

La población inicial es un elemento esencial al momento de plantear solución a un problema mediante un algoritmo genético, puesto que la selección de esta, influye de una manera determinante en el resultado final. Incluso hay algunas teorías que proponen que el tamaño debe ser variable con respecto al tiempo, evaluando de esta manera si un determinado cromosoma debe salir de la población al exceder un parámetro de tiempo de vida [35].

Un tamaño de población pequeño implica que los cromosomas o individuos pertenecientes a esta, pueden tener cierta tendencia a no cubrir adecuadamente el espacio de soluciones o de búsqueda que el problema de optimización requiera. Como contra parte, si se posee una población inicial de un tamaño considerable, es posible obtener soluciones que no converjan rápido representando inevitablemente tiempo computacional.

En el presente trabajo, se realiza una selección aleatoria de la población inicial abarcando un espacio de soluciones amplio, con el fin de no caer en una convergencia precoz hacia óptimos locales.

4.3.3 FUNCIÓN OBJETIVO

Básicamente, la función objetivo es una ecuación que determina que tan bueno o malo es un cromosoma en particular para la resolución del problema de optimización que se está tratando. Esta función por ende, permite evaluar a todos los individuos de la población dependiendo de su codificación inicial, generando un dato único que expresa el rendimiento o la bondad del dato con respecto a los demás.

Esta herramienta ayuda a cuantificar la población existente mediante la adjudicación de valores reales a cada cromosoma, permitiéndole al algoritmo genético, conocer cuáles son los datos que pueden llegar a ser la mejor solución del problema y por ende, ser estos más probables de ser seleccionados para la creación de una nueva generación, que se espera sea mejor que sus respectivos padres.

Para el desarrollo de este documento, la función objetivo o de costo es negativo del logaritmo de la verosimilitud marginal referenciada en la ecuación (11) (ver sección 4.2). Allí como se comentó anteriormente, se pretende encontrar los hiper-parámetros ($\sigma_f, \lambda, \sigma_n$) óptimos que minimicen dicha función de acuerdo a los datos de entrenamiento X existentes.

4.3.4 SELECCIÓN

El método de selección de individuos de un algoritmo genético es muy relevante para un adecuado rendimiento de este, ya que afecta directamente y de una manera significativa su convergencia. La filosofía de aplicación de una técnica de selección es la siguiente: "Mientras mejor sea la valoración de un individuo determinado (valor de su función objetivo), mayor probabilidad tendrá de permanecer en la población y procrearse o aparearse [36]".

El método usado en el presente proyecto, se denomina "*Selección por ruleta*", y fue seleccionado debido al hecho que es un método sencillo que no eleva en gran medida el costo computacional, y no posee parámetros adicionales que deban ser tenidos en cuenta para la selección de los individuos a recombinarse, como por ejemplo: la "*Selección por torneo*", en donde existe la necesidad de establecer la cantidad de individuos que formarán el subconjunto de selección [52].

En pocas palabras, la técnica de "*Selección por ruleta*" propone o asume que la probabilidad de selección de un individuo, es proporcional a su rendimiento o valoración [37]. Es decir, la probabilidad de seleccionar un i –ésimo individuo es equivalente:

$$p_i = \frac{w_i}{\sum_{i=1}^N w_i}, \quad (i = 1, 2, 3, \dots, N) \quad (12)$$

Donde w_i es el dato de valoración de i –ésimo perteneciente a la población de N cromosomas.

Considerando lo anterior, la técnica de *selección por ruleta* plantea en su teoría la generación de una línea recta imaginaria, en la cual los pesos p_i indicadores de la valoración de cada individuo de acuerdo a su función objetivo, se van posicionando a través de esta, creando diversos sectores o particiones que van a permitir seleccionar el individuo. Es relevante anotar, que un dato aleatorio X es escogido (uniformemente distribuido), con el fin de posicionarlo en dicha recta y aquella sección donde se ubique será el individuo elegido (ver Figura 5).

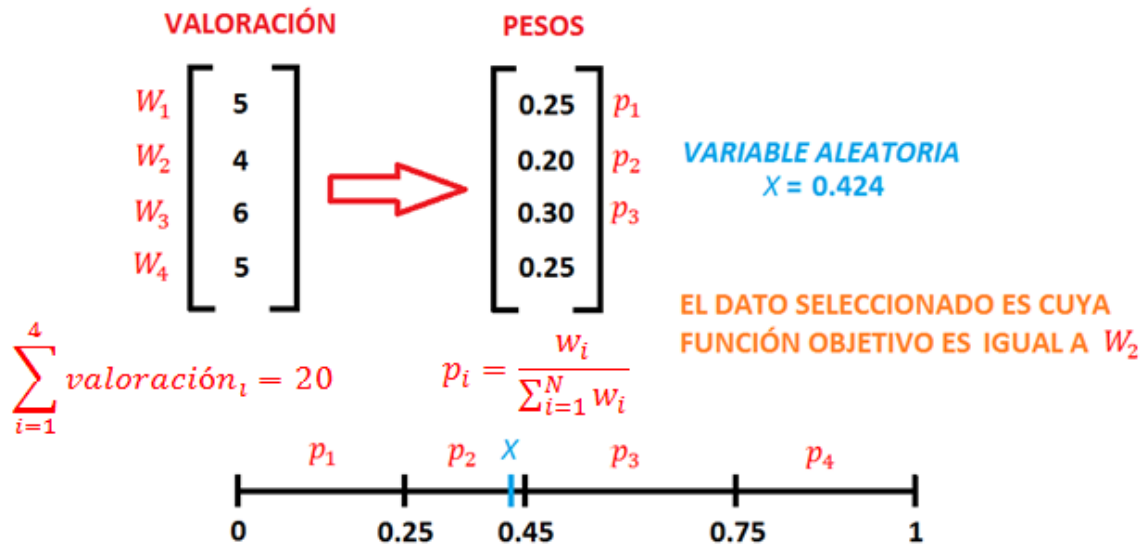


Figura 5. Ejemplo método de selección por ruleta

Bajo este procedimiento anterior, son seleccionados los dos padres que producirán un nuevo cromosoma en la población y de esta manera, identificar si mejora o no la solución al problema de optimización tratado, mediante la evaluación de la función objetivo. Se debe tener constante vigilancia de individuos que posean una función objetivo muy superior al resto, puesto que este hecho podría producir temprana convergencia en el algoritmo, generando que los nuevos individuos sean prácticamente copias de este que predomina sobre los demás.

4.3.5 RECOMBINACIÓN

La recombinación es una etapa esencial para el funcionamiento del algoritmo genético, puesto en esta instancia es donde se permite el intercambio de información genética de haciendo que los bloques constructivos de mejor calidad se repliquen a través de los individuos de la población y los que no posean una calidad tan alta disminuyan o incluso sean eliminados a través del paso de generaciones.

El tipo de recombinación usada en el proceso de optimización depende en gran medida del tipo de codificación que se esté usando en el problema. Considerando que el presente trabajo se basa en un algoritmo genético con codificación de valores reales, se hace relevante indicar que tipo de técnica se usa para efectuar la recombinación de los cromosomas, en pro de resolver el problema de optimización planteado (encontrar los hiperparámetros óptimos del Kernel que minimicen el negativo del logaritmo de la verosimilitud marginal, ver sección 4.2)

El método de *recombinación heurístico* como el aplicado en [39], es el escogido para efectuar la recombinación de cromosomas codificados con valores reales en el presente trabajo. Allí, se muestra el procedimiento de como generar dos descendientes a partir de los valores de la función objetivo de cada uno de sus padres. Para ello, se generan las algunas definiciones y se sigue la siguiente metodología:

Definiciones básicas:

Cromosoma del Padre 1 = P_1 Función objetivo Padre 1 = F_1

Cromosoma del Padre 2 = P_2 Función objetivo Padre 2 = F_2

Cromosoma del Hijo 1 = H_1 Función objetivo Hijo 2 = H_2

De allí se tiene que:

$$\text{si } \begin{cases} F_1 > F_2 \\ F_2 > F_1 \end{cases} \begin{cases} H_1 = P_1 \\ H_2 = P_1 \pm r * (P_1 - P_2) \end{cases} \quad \begin{cases} H_1 = P_2 \\ H_2 = P_2 \pm r * (P_2 - P_1) \end{cases} \quad (13)$$

Donde r es un dato aleatorio uniformemente distribuido entre 0 y 1 que va variando en cada iteración. Puesto que el cromosoma del Hijo 2 (H_2) representa un concepto de distancia o lejanía entre ambos padres, es posible afirmar que el valor r va a situar aleatoriamente al cromosoma del descendiente entre ellos, permitiendo diversidad en este nuevo cromosoma y evadiendo caer de manera inequívoca en óptimos locales [33].

Considerando la descripción anterior, se puede afirmar que siempre se va a tener que uno de los descendientes va a ser igual al padre de mejor función objetivo, mientras que el segundo hijo ocupará una nueva posición en el espacio global de soluciones (cromosoma ubicado entre ambos padres), siendo así una nueva alternativa de resultado.

En la figura 6, se puede verificar la conceptualización básica del método definido anteriormente para recombinaciones de cromosomas con codificación real.

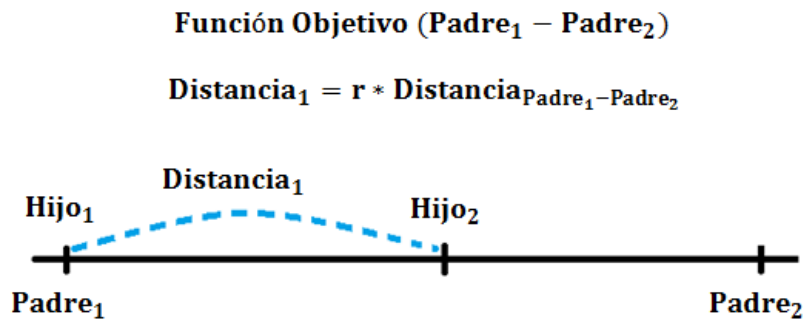


Figura 6. Conceptualización recombinación heurística.

La recombinación heurística busca por tanto, que los cromosomas que vayan evolucionando a través de cada iteración se encuentren cada vez más cerca entre sí, haciendo que la distancia numérica que los separa sea menor, en pro de buscar la convergencia.

La figura 7 presenta un ejemplo numérico referente al método de recombinación heurística para cromosomas con codificación de valor real. Allí se asume que la codificación de los individuos o padres (P_1 y P_2) está realizada mediante un único valor real, y que se tiene conocimiento de su respectiva función objetivo previamente evaluada (F_1 y F_2). Además se genera un dato aleatorio uniformemente distribuido equivalente a 0.75.

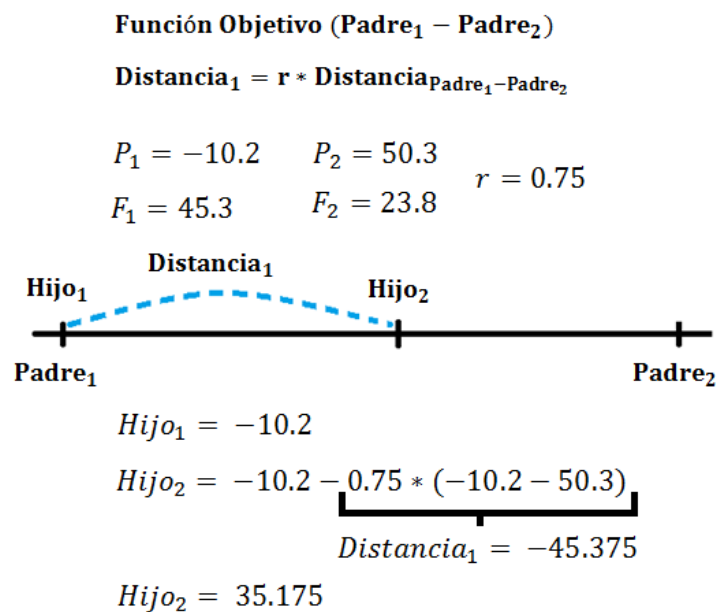


Figura 7. Ejemplo numérico recombinación heurística.

Es importante recalcar que en el presente trabajo, los individuos son representados mediante tres datos de valor real, equivalentes a los hiperparámetros de la función Kernel (σ_f , λ , σ_n), y por ende el procedimiento mostrado en la figura 7 se debe ejecutar para cada uno de ellos con el fin de generar un nuevo individuo.

4.3.6 MUTACIÓN

La mutación es considerada en el algoritmo genético un parámetro fundamental o básico, esto se debe a que diversos estudios en la literatura han expuesto que aunque el operador de recombinación es el encargado de encontrar en el espacio de búsqueda las distintas soluciones al problema de optimización, el operador de mutación fortalece dicha exploración puesto que permite la creación de información nueva que no se encontraba presente en la población inicial, además de buscar de manera permanente posibles soluciones en la vecindad de los individuos resultantes después de un proceso de recombinación [40,41].

Para codificaciones de valor real, la teoría propone la generación de un intervalo limitado de cambio [49], en el cual se tiene muy presente el número de la generación actual (número de iteraciones hasta el momento). En el inicio del procedimiento, el intervalo de cambio de los cromosomas para efectuar su mutación es amplio, a diferencia de cuando se va avanzando de generación en generación, donde este rango de cambio para los individuos va decreciendo.

Esta técnica de mutación, se conoce en la literatura como *mutación no uniforme*, y opera de la siguiente manera [43]:

$$Hijo\ mutado = \begin{cases} Hijo_1 + (Limite\ Superior - Hijo_1) * f(G) \\ Hijo_1 - (Hijo_1 - Limite\ Inferior) * f(G) \end{cases} \quad (14)$$

Donde $f(G)$ es la función de estabilidad gobernada por la generación actual en la que se desarrolla la mutación. G_{Max} es el máximo número de iteraciones permitidas (influye en el costo computacional), G es el valor de la generación actual y b es un parámetro de forma que para el desarrollo del trabajo tiene un valor de 1, puesto que se decide que la función $f(G)$ sea lineal.

$$f(G) = (r * \left(1 - \frac{G}{G_{Max}}\right))^b \quad (15)$$

Esta ecuación (15), tiene como objetivo principal irle restando peso al operador de mutación mientras más iteraciones hayan transcurrido, ya que a medida que aumenta la cantidad de generaciones también se espera que la población donde se encuentra los hiper-parámetros esté convergiendo y se aproxime a un mismo valor.

La figura 8 representa de manera gráfica la conceptualización de la mutación no uniforme propuesta para el presente trabajo. Es importante resaltar, que es arbitraria la decisión de aplicar la ecuación (14) hacia el límite superior o el inferior, el hecho fundamental es aplicar este operador y generarle una vecindad al cromosoma seleccionado.

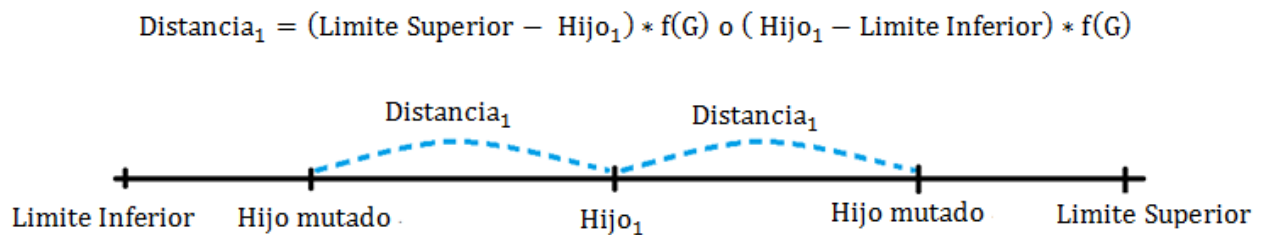


Figura 8. Conceptualización mutación no uniforme

Como se puede notar en la ecuación (15), al igual que en el proceso de recombinación, en la mutación no uniforme se usa el parámetro r (variable aleatoria uniformemente distribuida entre 0 y 1), para generar diversidad en el cálculo y evitar así óptimos locales en el problema.

Tanto el límite superior como el inferior deben ser seleccionados de una manera cautelosa, puesto que el hecho de generar un intervalo muy amplio entre ellos implica que el posible cromosoma intervenido difiera en alta medida del original, pudiéndose perder quizá información valiosa del cromosoma o individuo inicial.

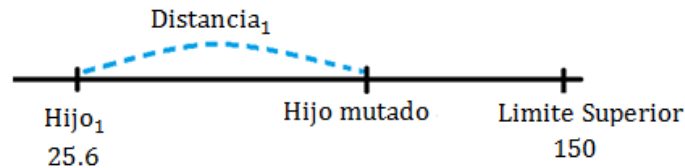
En la figura 9, se puede observar un ejemplo numérico de la mutación no uniforme para un cromosoma de valor real de una única variable, para el cual se ha seleccionado un número de generaciones máxima de 500, y se efectúa la mutación hacia el límite superior del intervalo con un valor de 150.

$$\text{Hijo mutado} = \text{Hijo}_1 + (\text{Límite Superior} - \text{Hijo}_1) * f(G)$$

$$\text{Distancia}_1 = (\text{Límite Superior} - \text{Hijo}_1) * f(G)$$

$$f(G) = \left(r * \left(1 - \frac{G}{G_{\text{Max}}} \right) \right)^b$$

$$\text{Limite Superior} = 150 \quad \text{Hijo}_1 = 25.6 \quad G = 125 \quad G_{\text{Max}} = 500 \quad r = 0.42 \quad b = 1$$



$$f(G) = 0.315$$

$$\text{Distancia}_1 = (150 - 25.6) * 0.315$$

$$\text{Distancia}_1 = 39.816$$

$$\text{Hijo mutado} = 25.6 + 39.816$$

$$\text{Hijo mutado} = 65.416$$

Figura 9. Ejemplo numérico mutación no uniforme.

Como se comentó en la sección 4.2, el presente trabajo busca encontrar de manera óptima los hiperparámetros de la función Kernel del proceso Gaussiano (σ_f , λ , σ_n), lo cual implica que los cromosomas usados posean tres datos de valor real en su interior, y por ende ante la necesidad de aplicar un operador de mutación no uniforme, el procedimiento debe hacerse en tres ocasiones, uno para cada uno de ellos.

5. MATERIALES Y MÉTODOS

5.1 DESCRIPCIÓN BASE DE DATOS

La base de datos usada para la representación de este proyecto fue extraída del "*Prognostics Data Repository*" que posee la NASA [26]. Esta colección de bases de datos ha sido suministrada por diferentes entidades, entre ellas: universidades, agencias y algunas compañías del sector industrial, con el fin de que sean desarrollados algoritmos de predicción y pronóstico.

Particularmente para este trabajo, se seleccionó el dataset denominado "*Battery*" de dicho repositorio, la cual brinda información detallada de una muestra de 36 baterías de Litio-ion tanto en sus procesos de operación ("*carga*", "*descarga*"), como en la medición de algunos parámetros físicos de la misma dados bajo el nombre de "*impedancia*".

En la figura 10, es posible observar la estructura que poseen los datos para cada uno de sus procesos de operación:

A continuación, se explica a que hacen referencia cada una de las series de tiempo de los diferentes ciclos de la batería. Cabe resaltar que se decide nombrar dichas variables en idioma inglés, con el fin de respetar la nomenclatura inicial suministrada originalmente por la base de datos [26].

CICLO DE CARGA

- *Voltage_Measured*: Este parámetro hace referencia a la diferencia de potencial existente en los bornes de la batería, y está dada en Voltios (V).
- *Current_Measured*: Esta variable mide la corriente de salida de la batería, su valor está en Amperios (A).
- *Temperature_Measured*: Entrega la temperatura de la batería al cual se le está haciendo la prueba. Su valor está en grados Celsius (°C).
- *Current_Charge*: Esta variable hace referencia a la corriente medida en el cargador mientras se encuentra en el periodo de carga. Su valor está en Amperios (A).
- *Voltage_Charge*: Este parámetro mide la tensión existente en el cargador cuando está cargando. Su valor se da en Voltios (V).
- *Time*: Es un vector que contiene el momento cada uno de los instantes en el cual se adquieren los datos durante el proceso de carga. La variable Time es un arreglo de $1 \times n$, donde n es el número de datos existentes en el proceso.

CICLO DE DESCARGA

- *Voltage_Measured*: Este parámetro hace referencia a la diferencia de potencial existente en los bornes de la batería, y está dada en Voltios (V).
- *Current_Measured*: Esta variable mide la corriente de salida de la batería, su valor está en Amperios (A).
- *Temperature_Measured*: Entrega la temperatura de la batería al cual se le está haciendo la prueba. Su valor está en grados Celsius (°C).
- *Current_Charge*: Esta variable hace referencia a la corriente medida en la carga mientras se encuentra en el proceso de descarga. Su valor está en Amperios (A).
- *Voltage_Charge*: Este parámetro mide la tensión existente en la carga cuando el cargador se encuentra suministrándole energía. Su valor se da en Voltios (V).
- *Time*: Es un vector que contiene el momento cada uno de los instantes en el cual se adquieren los datos durante el proceso de carga. La variable Time es un arreglo de 1xn, donde n es el número de datos existentes en el proceso.
- *Capacity*: Su valor está dado en Amperio-hora (Ah), y hace alusión a la capacidad de la batería para poder generar su ciclo de descarga.

Es relevante acotar, que para los ciclos de impedancia descritos en [26], no se presenta una explicación para cada uno de los parámetros extraídos.

Las series de tiempo para cada una de las variables nombradas anteriormente, fueron obtenidas por medio del procedimiento que se describe a continuación:

"A una *batería de litio-ion*, se le realizan 3 perfiles operativos diferentes (carga, descarga y de impedancia) a *cierta temperatura*. El *ciclo de carga*, se lleva a cabo con una corriente constante (CC) de 1.5A hasta que el voltaje de la batería sea de 4.2V, y posteriormente se continua con un voltaje constante hasta que la *corriente de carga* se reduzca 20 mA. En el *ciclo de descarga* se conecta un elemento tal que le consuma una corriente fija a la fuente (2A, 4A , Onda cuadrada 4A con 50% de ciclo útil), y se le permite a la batería descender hasta un *voltaje final determinado* (2V , 2.2V , 2.5V , 2.7 V). Ahora bien, La medición de impedancia se lleva a cabo a través de una espectroscopia de impedancia electroquímica (EIS) con un barrido de frecuencias de 0,1 Hz a 5 kHz [42].

"Los ciclos de carga y descarga repetidas dan lugar a un envejecimiento acelerado de las baterías, mientras que las mediciones de impedancia proporcionan información sobre los parámetros de la batería interna que cambian a medida que avanzan los diferentes ciclos. Los experimentos se detienen cuando se detecta una disminución en la capacidad nominal de la batería [43]".

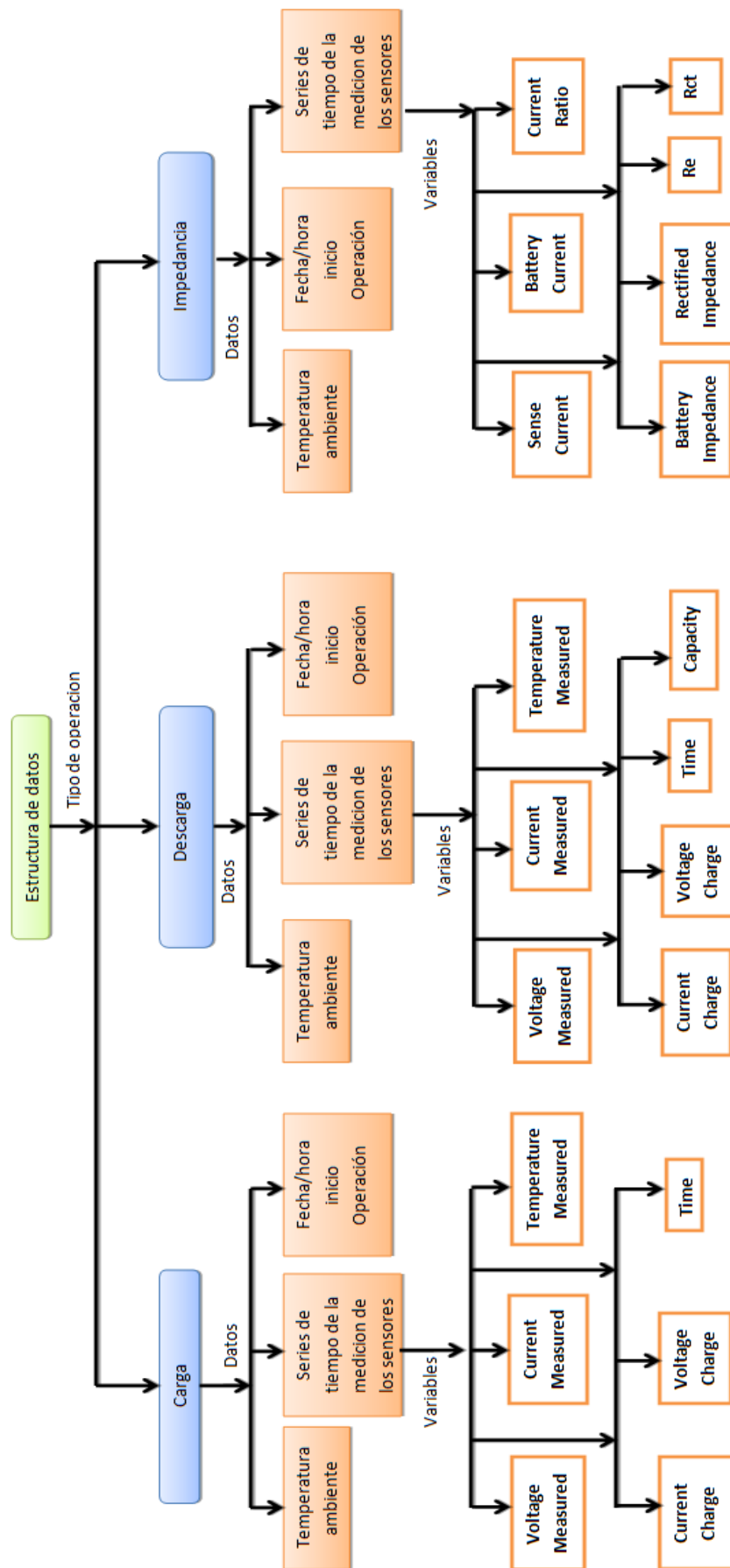


Figura 10. Estructura jerárquica de la base de datos "Battery" del Prognostics Data Repository de la NASA.

5.2 CARACTERIZACIÓN BASE DE DATOS

La caracterización de la base de datos, es un ítem fundamental para poder atacar el problema de una manera efectiva. Aquí, se busca obtener la mayor cantidad de información posible de la base de datos de trabajo mediante la menor cantidad de variables posibles, pero que me intenten representar toda la variabilidad que tiene el problema, y no haya pérdida de mucha información.

Por ejemplo, en [44], teniendo en la base de datos las medidas básicas de corriente, voltaje, y demás, se generan 5 nuevas variables que sean indicativas de la capacidad de carga, para poder desde allí aplicar técnicas de regresión y encontrar la relación de estas últimas con el tiempo de vida útil restante (RUL). A su vez, en [45] usan únicamente los valores de corriente de la batería en su estado de carga, y los valores de tensión o diferencia de potencial del dispositivo en su fase de descarga para efectuar la caracterización de la base de datos.

Ahora bien, para realizar la caracterización de la base de datos del presente trabajo [26], se ejecutó un procedimiento similar al efectuado en [46], en donde se tiene únicamente en cuenta la información del ciclo de descarga de cada una de las baterías, ya que este tipo de proceso posee datos fundamentales para poder determinar finalmente la capacidad (Ampere-hora) de la batería en estudio. Finalmente, se eligieron 15 parámetros que permitieran generar una buena caracterización o parametrización de la base de datos; en su gran mayoría, estas variables representan la cuantificación de los momentos de orden k (varianza, asimetría, curtosis) de las series de tiempo para el voltaje y la temperatura en la batería en estudio.

Se hace relevante anotar, que en estos 15 parámetros descritos con antelación también está incluida la variable regresora, que describe la capacidad actual con la que cuenta la batería para ese instante de tiempo específico. A continuación, se muestran de manera explícita como se desarrolló la parametrización de la base de datos de las baterías de Litio-ion.

1) Se determina cuantos ciclos de descarga hay en toda la estructura de datos de la batería. Es decir, se examina cuantos ciclos de carga, impedancia y descarga existen, y se tiene en cuenta la cantidad de veces que se desarrolló este último proceso. Finalmente este número de ciclos de descarga se denota con la letra n .

2) Se Parametrizan cada uno de los ciclos de descarga por medio de las 15 variables descritas a continuación:

2.1) Índice de fluctuación para el voltaje de descarga:

$$IFV = \frac{\sqrt{\sum_{i=1}^n (V_i - \bar{V})^2}}{\omega} \quad (16)$$

V_i = Dato i – esimo de los n datos de voltaje del ciclo de descarga.

\bar{V} = Promedio de los datos de voltaje en el ciclo de descarga.

ω = Frecuencia de muestro de los datos de voltaje para el ciclo de descarga.

2.2) Índice de fluctuación para la temperatura:

$$IFT = \frac{\sqrt{\sum_{i=1}^n (T_i - \bar{T})^2}}{\omega} \quad (17)$$

T_i = Dato i – esimo de los n datos de temperatura del ciclo de descarga.

\bar{T} = Promedio de los datos de temperatura en el ciclo de descarga.

ω = Frecuencia de muestro de los datos de temperatura en el ciclo de descarga.

2.3) Índice de asimetría para el voltaje de descarga:

$$IAV = \frac{\sum_{i=1}^n (V_i - \bar{V})^3}{\sigma^3} \quad (18)$$

V_i = Dato i – esimo de los datos de voltaje del ciclo de descarga.

\bar{V} = Promedio de los datos de voltaje en el ciclo de descarga.

σ = Desviación estándar de los datos de voltaje en el ciclo de descarga.

2.4) Índice de asimetría para la temperatura:

$$IAT = \frac{\sum_{i=1}^n (T_i - \bar{T})^3}{\sigma^3} \quad (19)$$

T_i = Dato i – esimo de los datos de temperatura del ciclo de descarga.

\bar{T} = Promedio de los datos de temperatura en el ciclo de descarga.

σ = Desviación estándar de los datos de temperatura en el ciclo de descarga.

2.5) Índice de curtosis para el voltaje de descarga:

$$IKV = \frac{\sum_{i=1}^n (V_i - \bar{V})^4}{\sigma^4} \quad (20)$$

V_i = Dato i – esimo de los datos de voltaje del ciclo de descarga.

\bar{V} = Promedio de los datos de voltaje en el ciclo de descarga.

σ = Desviación estándar de los datos de voltaje en el ciclo de descarga.

2.6) Índice de curtosis para la temperatura:

$$IKT = \frac{\sum_{i=1}^n (T_i - \bar{T})^4}{\sigma^4} \quad (21)$$

T_i = Dato i – esimo de los datos de temperatura del ciclo de descarga.

\bar{T} = Promedio de los datos de temperatura en el ciclo de descarga.

σ = Desviación estándar de los datos de temperatura en el ciclo de descarga.

2.7) Valor máximo de la serie de voltaje en el ciclo de descarga:

V_{max} = Indica el valor de tensión más alto en la serie de tiempo del voltaje.

2.8) Tiempo del valor máximo de la serie de voltaje en el ciclo de descarga:

t_{max} = Instante en el cual se produce el valor del parámetro V_{max} .

2.9) Valor mínimo de la serie de voltaje en el ciclo de descarga:

V_{min} = Indica el valor de tensión más bajo en la serie de tiempo del voltaje.

2.10) Tiempo del valor mínimo de la serie de voltaje en el ciclo de descarga:

t_{min} = Instante en el cual se produce el dato del parámetro V_{min} .

2.11) Valor máximo de la serie de temperatura en el ciclo de descarga:

T_{max} = Indica el valor de temperatura más alto en el ciclo de descarga.

2.12) Tiempo del valor máximo de la serie de temperatura en el ciclo de descarga:

t_{Tmax} = Instante en el cual se produce el valor del parámetro T_{max} .

2.13) Valor mínimo de la serie de temperatura en el ciclo de descarga:

T_{min} = Indica el valor de temperatura más bajo para la serie descrita.

2.14) Tiempo del valor mínimo de la serie de temperatura en el ciclo de descarga:

t_{Tmin} = Instante de tiempo para el cual se produce la temperatura más baja.

2.15) Capacidad de la batería

Cap = Capacidad existente de la batería para un ciclo de descarga específico.

Los parámetros designados desde el 2.1 hasta el 2.14 se denominan variables de entrada y son aquellas que ingresan al modelo de regresión, con el fin de que sea estimado con cierto intervalo de confianza el dato del parámetro 2.15 o variable regresora (Capacidad).

Cabe anotar además, que se decide tener en cuenta las variables de voltaje y temperatura para caracterizar la batería, puesto que se conoce que son parámetros críticos y de gran variación con respecto al tiempo, por lo cual son esenciales al momento de determinar la capacidad en Amperes-hora (Ah) que posee una batería de Litio ion [47].

3) Generación matriz de variables de entrada y salida para el proceso Gaussiano

Como último procedimiento de la parametrización de la base de datos, se condensa toda la información calculada en una matriz entrada - salida de dimensión $n \times p$, donde como se dijo anteriormente, n representa la cantidad de ciclos de descarga existentes en la estructura de datos inicial, y p hace alusión a cada uno de los 15 parámetros computados para cada uno de los ciclos.

En la figura 11, es posible verificar la distribución de la matriz entrada-salida usada en el presente proyecto. En este gráfico, el subíndice de cada parámetro representa la i -ésima vez que se detecta un ciclo de descarga (ciclo i -ésimo de los n ciclos de descarga existentes).

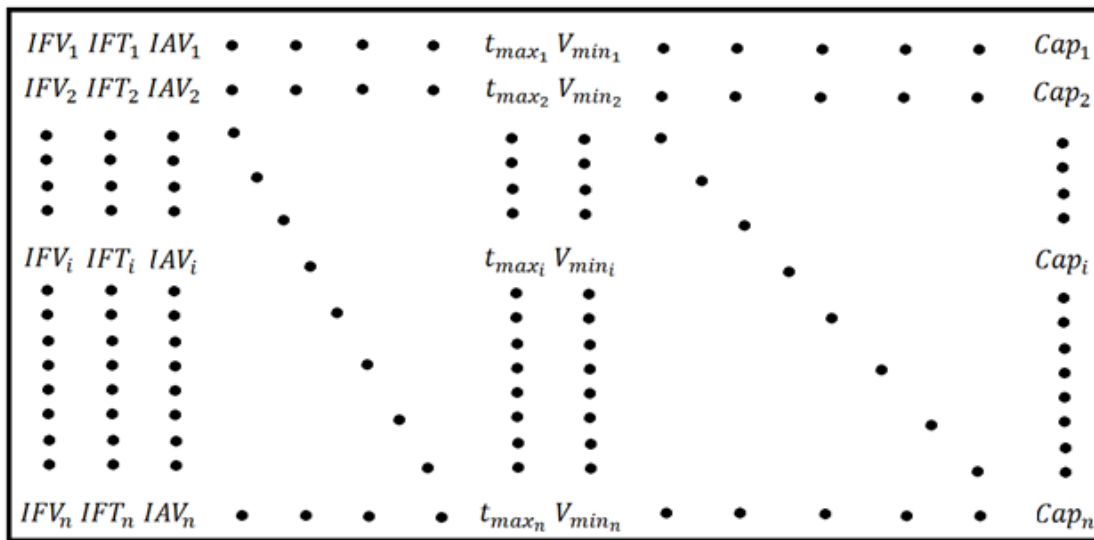


Figura 11. Matriz Entrada/Salida para la estimación del tiempo de vida útil restante de las baterías de Litio-ion.

Para contextualizar un poco más, IKV_1 e IKT_1 representan los índices de curtosis para los datos de voltaje y temperatura obtenidos en el primer ciclo de descarga; a su vez, IAV_i e IAT_i relacionan los índices de asimetría para los datos de voltaje y temperatura del i-ésimo ciclo de descarga. Este tipo de nomenclatura se extiende para cada uno de los 15 parámetros seleccionados con el fin de conformar en su totalidad la matriz de entrada-salida del modelo.

5.3 VALIDACIÓN

Considerando que ya se tiene una matriz compacta en la cual se encuentran condensados los parámetros que representan o caracterizan la base de datos inicial, se efectuó la división de esta en dos grupos característicos, los cuales se explican a continuación:

1) *Grupo de entrenamiento (Training Set)*: Este grupo fue sumamente importante para la obtención del tiempo de vida útil restante de las baterías de Litio-Ion, puesto que los datos pertenecientes a este, fueron los encargados de construir el modelo matemático que permitía estimar el valor del parámetro de salida (Capacidad en Ah) para la cantidad de variables de entrada existentes.

2) *Grupo de Prueba (Test Set)*: Dicha porción de la matriz, permitió cuantificar qué tan bien se ajusta el modelo creado a aquellos datos que no tuvieron ningún tipo de aporte al establecimiento del mismo.

Se hace relevante comentar, que para este caso específico en el cual se está usando un proceso Gaussiano para efectuar una regresión, se realiza una partición a la matriz de parámetros equivalente al 70%, donde este porcentaje representa la cantidad de datos usados como grupo de entrenamiento, y el 30% restante es usado para efectuar la validación del modelo [49,50].

En la figura 12, se muestra un ejemplo de la división generada en la matriz de parámetros para la selección de los datos de entrenamiento (primeras k-ésimas muestras) y las filas restantes que son usados como datos de prueba del modelo. No está de más recalcar que demarcados en rojo se encuentran los datos de entrenamiento (Training Set) y en azul los datos de prueba (Test Set).

Es esencial comentar, además, que la última columna de la matriz Entrada/Salida corresponde a la capacidad en Ah en la batería de Litio-Ion, la cual hace referencia a la variable regresora o de salida del modelo propuesto.

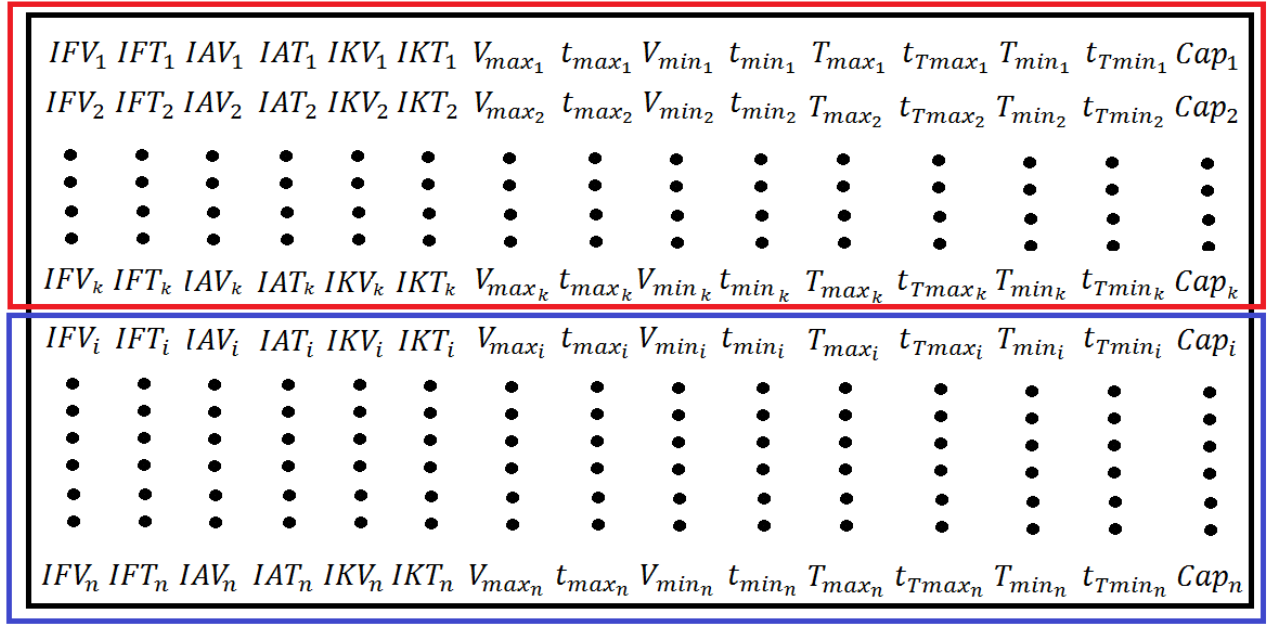


Figura 12. División de la matriz de parámetros para el Training Set y Test Set.

5.4 REGRESIÓN PARA HALLAR EL RUL

Para encontrar la estimación del tiempo de vida útil de las baterías de Litio-Ion, se hizo fundamental tomar como base un Toolbox de Matlab que contiene todas las pautas y códigos probados del libro de Rasmussen y Williams [19], el cual es un libro pilar en el mundo del aprendizaje de máquina aplicado. Lo anterior, garantiza una perfecta implementación y desarrollo de cálculos en todo lo que implica usar un proceso Gaussiano para resolver un determinado problema. Como se comentó en la sección 4.2, este tipo de herramientas tienen como particularidad, que implementan todo su proceso de optimización basado en métodos que usan gradiente.

Es válido recalcar nuevamente, que este tipo de métodos de optimización tienden a caer en resultados ligados a mínimos locales, ya que dependen en gran medida del punto inicial suministrado. Por lo cual, se decide implementar la estimación de hiper-parámetros del Kernel (σ_f , λ y σ_n) mediante una técnica heurística que permita recorrer todo el espacio de soluciones, como es el caso de un algoritmo genético de valor real.

Teniendo en cuenta que el Kernel de un proceso Gaussiano busca generar una transformación al espacio de variables contenidas en la matriz de entradas y salidas, permitiendo así encontrar una correlación entre los distintos puntos contenidos en dicha matriz, se hizo intuitivo pensar en seleccionar una covarianza o Kernel exponencial

cuadrado, ya que dicha función representada en la ecuación (7) indica que puntos que se encuentren cerca en el espacio de entrada están muy correlacionados, y que dicha relación va disminuyendo a medida que se alejan (es conveniente un conocimiento previo de la función de covarianza, tal como se trata en el contexto de [48]). Y este es el caso, de las baterías de Litio-ion, donde se espera que probablemente ciclos de descargas contiguos estén muy correlacionados.

Además de lo anterior, y con el fin objetivo principal de realizar una validación del proceso desarrollado (Proceso Gaussiano + Algoritmo genético de valor real), el Kernel seleccionado es uno exponencial cuadrado ya que es usado también en el desarrollo de [9], y de esta manera, se pueden realizar diversos símiles usando herramientas estadísticas como lo es: el error cuadrático medio para verificar el sesgo entre los datos reales y sus respectivas estimaciones.

Esta unidad del trabajo busca mostrar y exponer de manera explícita, el procedimiento propuesto para el cálculo del RUL de las baterías de Litio-ion contenidas en la base de datos [26].

Considerando que los ciclos de descarga de las baterías analizadas son las que guardan la información más relevante del estado actual del dispositivo, se efectúa en primera instancia, la clasificación de este tipo de ciclos para cada uno de los elementos contenidos en la base de datos. Este resultado puede ser verificado en la sección resultados (ver Tabla 1).

Luego de tener la subdivisión de cada una las baterías mediante sus tipos de ciclo (carga, descarga e impedancia), se procede con la etapa de caracterización de los ciclos de descarga encontrados en esta clasificación. Allí, se siguen las pautas de parametrización descritas en la sección 5.2 (caracterización de la base de datos), en donde se tendrán 15 variables (14 de entrada y 1 de salida), con las que se pretende tener la información suficiente para explicar el ciclo en particular. Finalmente, se tendrá una matriz de entrada/salida para cada batería de Litio-ion con n ciclos de descarga, donde a cada uno de ellos le corresponderán p variables asignadas ($p = 15$) debido a la parametrización realizada (Ver figura 11).

De acuerdo a la figura 12, y tal como se había comentado, para el presente trabajo se presenta una división de la matriz de entrada/salida en donde el 70% de los datos serán usados para el grupo de entrenamiento y del 30% restante para el grupo de Test o verificación del modelo. Se tendrá entonces destinado para el *Training Set* (m ciclos de descarga) y como *Test Set* ($n - m$ ciclos), donde cada ciclo de descarga

para cualquier grupo, tendrá de igual manera sus p variables asignadas por la caracterización efectuada anteriormente.

Luego de tener definidos los grupos de entrenamiento y validación, es esencial considerar qué tipo de operaciones se deben realizar con estos datos. Para ello, se debe tener en cuenta de manera inicial, el valor de la verosimilitud marginal que básicamente representa la función de costo a optimizar (ecuación (11)). Teniendo en cuenta lo nombrado en la sección 4.2, se discrimina a continuación la definición de cada variable de la ecuación (11), con sus respectiva dimensión para poder efectuar los cálculos correspondiente:

$$\log p(y|X) = -\frac{1}{2}y^T(K(X,X) + \sigma_n^2 I)^{-1}y - \frac{1}{2}\log|K(X,X) + \sigma_n^2 I| - \frac{n}{2}\log 2\pi. \quad (11)$$

Discriminación y dimensionamiento de variables:

$$K(X_i, X_j) = \sigma_f^2 e^{\left(-\frac{|X_i - X_j|^2}{2\lambda^2}\right)} \rightarrow \text{Dato } (i, j) \text{ de la matriz } K(X, X).$$

$K(X_i, X_j)$ = Kernel exponencial cuadrado con hiperparámetros σ_f y λ .

y = Datos de la capacidad de la batería para el Training Set $\rightarrow (m \times 1)$.

X = Training Set sin dato de capacidad de la batería $\rightarrow m \times (p - 1) \rightarrow m \times 14$.

X^* = Test Set sin dato de capacidad de batería $\rightarrow (n - m) \times (p - 1) \rightarrow (n - m) \times 14$.

$\sigma_n^2 I$ = Matriz escalar de ruido con el hiperparámetro σ_n $\rightarrow (m \times m)$.

n = Número de ciclos de descarga de la batería.

Luego de efectuar el cálculo de la función de verosimilitud marginal $\log p(y|X)$ con las variables nombradas anteriormente, se obtendrá un dato escalar que representa aquel valor que se debe minimizar mediante la selección de unos adecuados hiper-parámetros (σ_f , λ y σ_n), puesto que los demás componentes de la función (11) no poseen ninguna variación. Es decir, los valores de σ_f , λ y σ_n que minimicen esta función de costo $\log p(y|X)$, serán los hiperparámetros seleccionados para efectuar la estimación de la capacidad de las baterías para el *Test Set* o *grupo de validación*.

Como se comentó anteriormente, el presente proyecto propone efectuar la optimización de esta función objetivo mediante el uso de un algoritmo genético de valor real, con el fin de buscar en un espacio de soluciones más amplio y evitar caer en mínimos locales.

5.4.1 OPTIMIZACIÓN DE LOS HIPER-PARÁMETROS MEDIANTE ALGORITMO GENÉTICO DE VALOR REAL

El algoritmo genético de valor real, es un método heurístico que posee diversos parámetros que pueden ser variados para modificar el desempeño de este, como es el caso de la población inicial y la cantidad de generaciones en las cuales se correrá el algoritmo de optimización. Para el actual trabajo, se decide seleccionar una población inicial de 500 individuos cuyo número de generaciones equivale también al mismo valor (Ver sección 6.2.1), es decir, inicialmente se tendrían 500 individuos aleatorios donde cada uno de ellos posee tres características equivalentes a los tres hiperparámetros σ_f , λ y σ_n que se desean optimizar. Esta premisa se puede visualizar de mejor manera en la figura 13.

POBLACIÓN			
Individuo 1	σ_{f1}	λ_1	σ_{n1}
Individuo 2	σ_{f2}	λ_2	σ_{n2}
•	•	•	•
•	•	•	•
•	•	•	•
Individuo j	σ_{fj}	λ_j	σ_{nj}
•	•	•	•
•	•	•	•
•	•	•	•
Individuo 500	σ_{f500}	λ_{500}	σ_{n500}

Figura 13. Población inicial para estimación de los hiperparámetros

Posteriormente de definir esta población, se requiere evaluar cada uno de estos individuos en la función objetivo $\log p(y|X)$ para determinar qué resultado entregan y verificar de que tan buena calidad es su respuesta. Es importante tener presente en cada iteración, qué individuo representa la mejor y la peor función objetivo para cuestiones de reemplazo a través del paso de las generaciones. Finalmente, se tendría un esquema como el siguiente:

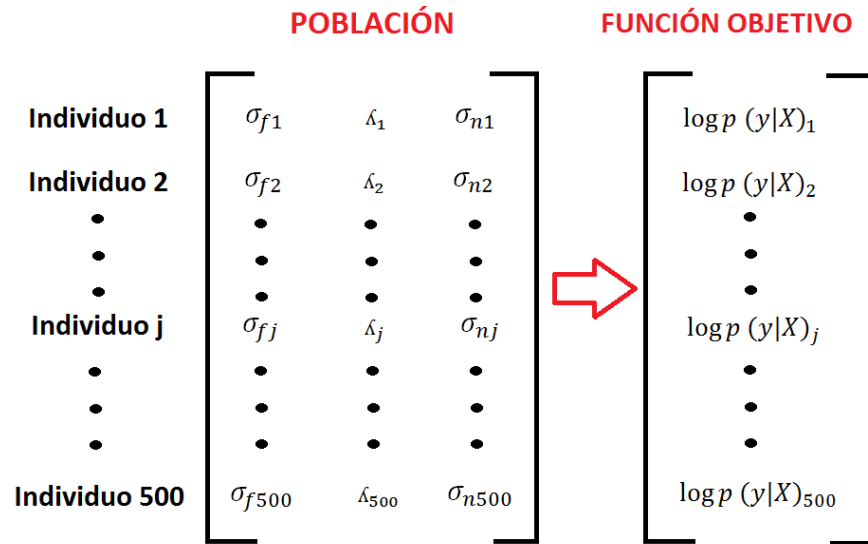


Figura 14. Población de individuos con su respectiva función objetivo

A continuación, sigue una de las etapas más importantes del proceso de estimación de los hiperparámetros del algoritmo genético, y es la fase de selección de padres. Para este caso puntual, se seleccionan dos individuos de los 500 existentes en la población con la finalidad de generar un hijo con características tales, que permita mejorar la genética de la población, y por ende obtener un valor menor en la función de costo $\log p(y|X)$. La selección se realiza de acuerdo al procedimiento mostrado en la sección 4.3.4, en donde un número aleatorio uniformemente distribuido entre 0 y 1 selecciona en primera instancia a uno de los padres, y posteriormente se vuelve a ejecutar para escoger el segundo progenitor.

En el tratamiento del presente trabajo, es relevante recordar que se decidió tener una tasa de recombinación equivalente a 1, lo cual significa que siempre que se ejecute el proceso de selección habrá un hijo de los padres elegidos.

La etapa de recombinación como tal, es también sumamente relevante en el proceso de optimización mediante un algoritmo genético, puesto que es esencialmente allí donde se producen los nuevos individuos para la población. El procedimiento de recombinación se efectúa de acuerdo a lo descrito en la sección 4.3.5, en donde se describe cómo generar un nuevo individuo teniendo genotipos de índole real mediante el método de recombinación heurística. A continuación, se describe de una manera más puntual la ecuación (13), la cual pretende exponer como desarrollar el proceso de recombinación para dos padres con genotipo de valor real. Se define además cada variable de la ecuación:

$$si \begin{cases} F_1 > F_2 \\ F_2 > F_1 \end{cases} \begin{cases} H_1 = P_1 \\ H_2 = P_1 \pm r * (P_1 - P_2) \\ H_1 = P_2 \\ H_2 = P_2 \pm r * (P_2 - P_1) \end{cases} \quad (13)$$

Donde:

P_1 = Hiperparámetros del padre 1 [σ_{f1} , λ_1 y σ_{n1}]

P_2 = Hiperparámetros el padre 2 [σ_{f2} , λ_2 y σ_{n2}]

F_1 = Función objetivo $\log p(y|X)$ evaluada en P_1

F_2 = Función objetivo $\log p(y|X)$ evaluada en P_2

H_1 = Hiperparámetros del Hijo 1 [σ_{fi} , λ_i y σ_{ni}]

H_2 = Hiperparámetros del Hijo 2 [σ_{fk} , λ_k y σ_{nk}]

r = Valor aleatorio que permite la geneación de un descendiente

Se hace esencial recalcar, que la expresión $F_1 > F_2$ significa que el padre P_1 posee una mejor función objetivo $\log p(y|X)$ que el padre P_2 , y viceversa para el caso $F_2 > F_1$.

Luego de esta etapa, se tendrán entonces dos nuevos individuos o hijos. Aunque hay algunos algoritmos genéticos que tienen en cuenta los dos descendientes obtenidos para continuar con el proceso de optimización; en el presente trabajo, se elimina uno de ellos de manera aleatoria con el fin de no sobrecargar computacionalmente el proceso de cálculo.

Es aquí, donde luego de tener el descendiente del proceso de recombinación, se establece una tasa de mutación específica para el proceso, en este caso puntual se dicta un valor de 0.1, lo cual quiere decir, que cuando finaliza el proceso de recombinación, se genera un número aleatorio L , el cual si $L < 0.1$ es necesario desarrollar la etapa de mutación para el nuevo individuo descrita en la sección 4.3.6, caso contrario los hiperparámetros del descendiente permanecerán constantes para la próxima instancia del proceso de optimización que se viene llevando.

Si la operación de mutación debe realizarse debido a que $L > 0.1$, el presente trabajo propone efectuar la operación descrita en la ecuación (14), Allí :

$$Hijo\ mutado = \begin{cases} Hijo_1 + (Limite\ Superior - Hijo_1) * f(G) \\ 0 \\ Hijo_1 - (Hijo_1 - Limite\ Inferior) * f(G) \end{cases} \quad (14)$$

Donde:

$Hijo_1$ = Descendiente con hiperparámetros $[\sigma_{fi}, \lambda_i \text{ y } \sigma_{ni}]$ luego de la recombinación.

Limite Superior = Valor máximo que toman los hiperparámetros $[\sigma_f, \lambda \text{ y } \sigma_n]$

$$f(G) = (r * \left(1 - \frac{G}{G_{Max}}\right))^b$$

Como se comentó en la sección 4.3.6, $f(G)$ es una función que permite que a medida que pasen las iteraciones, el proceso de mutación tenga un menor peso ya que se espera que la población de los hiper-parámetros se encuentre convergiendo, las variables de esta ecuación son las siguientes:

G_{Max} = Número máximo de generaciones en el algoritmo genético.

G = Generación actual del proceso.

r = Dato aleatorio uniformemente distribuido entre 0 y 1.

b = Parámetro de forma del modelo.

El resultado de la operación anterior, genera un nuevo individuo con un genotipo modificado, inicialmente con hiperparámetros $[\sigma_{fi}, \lambda_i \text{ y } \sigma_{ni}]$, y de manera final con valores $[\sigma_{fk}, \lambda_k \text{ y } \sigma_{nk}]$. En el presente proyecto, es relevante decir que se eligen límites superiores iguales a 100 para cada uno de los tres hiperparámetros a estimar ($\sigma_f, \lambda \text{ y } \sigma_n$). Además de lo anterior, se tendría un valor de $G_{Max} = 500$, considerando que este es el número máximo de iteraciones de la población. Finalmente, se selecciona un parámetro de forma $b = 1$ para cumplir con el objetivo de la función $f(G)$ sea lineal.

La última instancia de algoritmo genético de valor real propuesto, consiste en comparar la función objetivo o de costo $\log p(y|X)$ para el $Hijo_1$ ($\sigma_{fi}, \lambda_i \text{ y } \sigma_{ni}$) dado el caso que no se haya hecho el proceso de mutación, o para el *Hijo mutado* ($\sigma_{fk}, \lambda_k \text{ y } \sigma_{nk}$) en caso contrario. Básicamente, si en la evaluación realizada a este nuevo individuo en la función $\log p(y|X)$ se obtiene un mejor resultado, este reemplazará en la población a aquel que posea el peor rendimiento, y nuevamente será ejecutado el proceso de selección en la nueva iteración o generación. Esta acción se puede chequear en la figura 15.

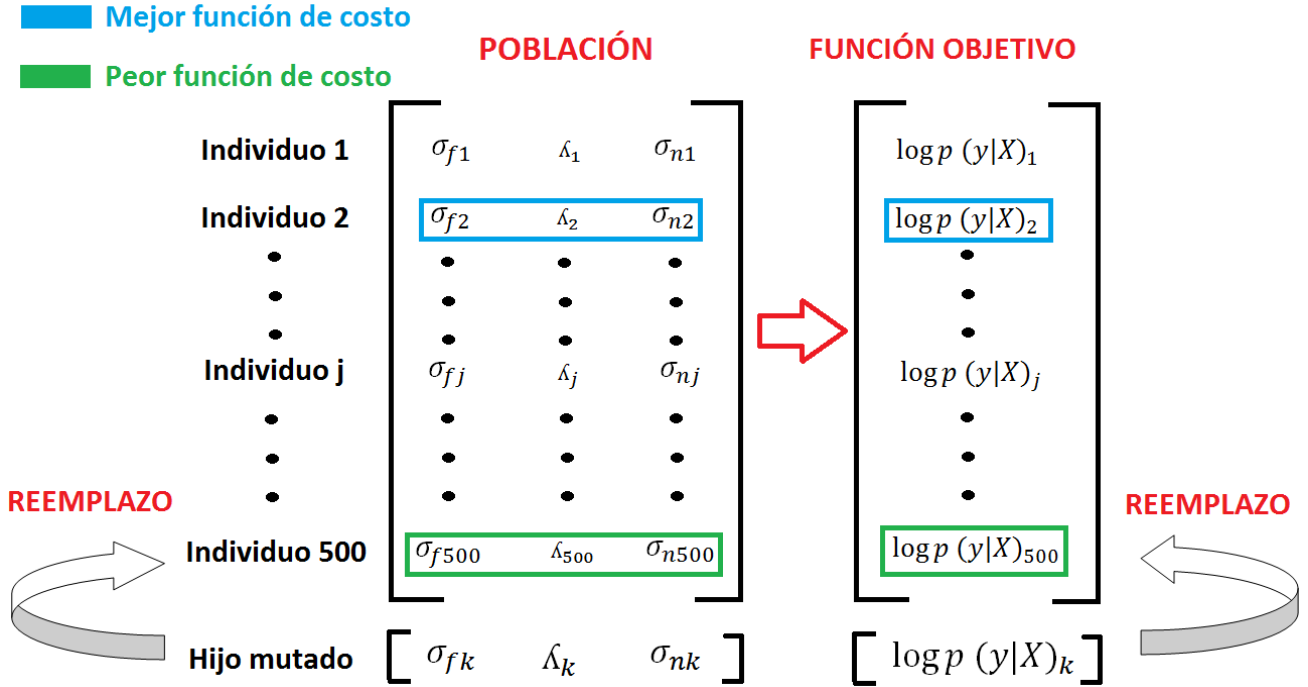
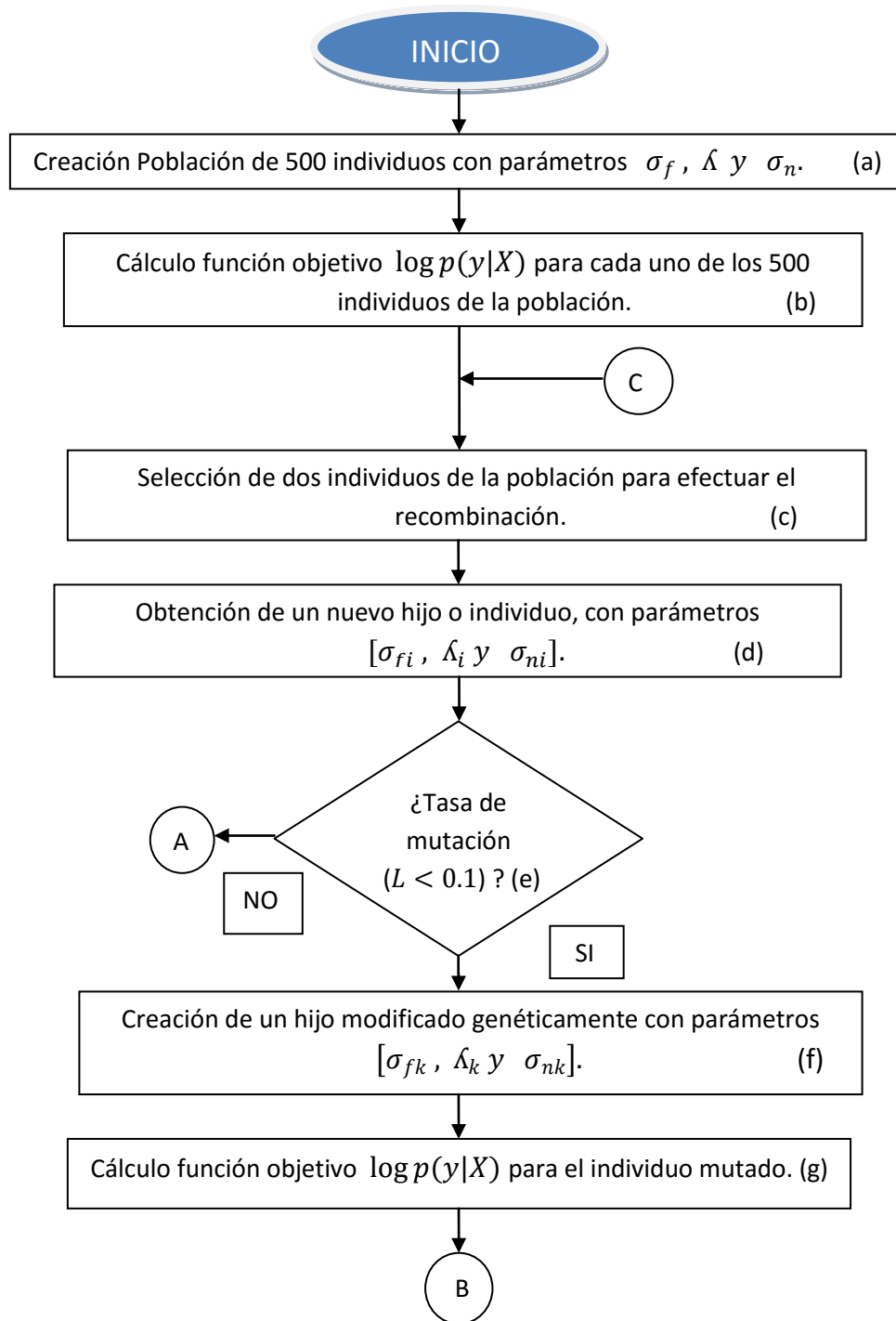


Figura 15. Reemplazo de individuos en población del algoritmo genético de valor real.

Finalmente, luego de que las 500 iteraciones sean efectuadas, se tendrá un individuo que representa el conjunto de hiperparámetros σ_f , λ y σ_n tales que minimizan la función de costo $\log p(y|X)$ a través de todo el proceso. Estos valores son en conclusión los hiperparámetros que permitirán la evaluación de las funciones de estimación de la media y la varianza (\bar{f}^* y $cov(f^*)$ respectivamente) de la capacidad de las baterías de Litio-ion en el *Test Set* o *Grupo de Prueba*.

La figura 16 representa un diagrama de flujo que sintetiza el proceso efectuado en el presente trabajo para el cálculo de la capacidad de las baterías de Litio-ion y por ende su respectivo RUL. Posteriormente y con ayuda de la nomenclatura interna del diagrama, se hace una breve explicación de lo que representa cada bloque de este.



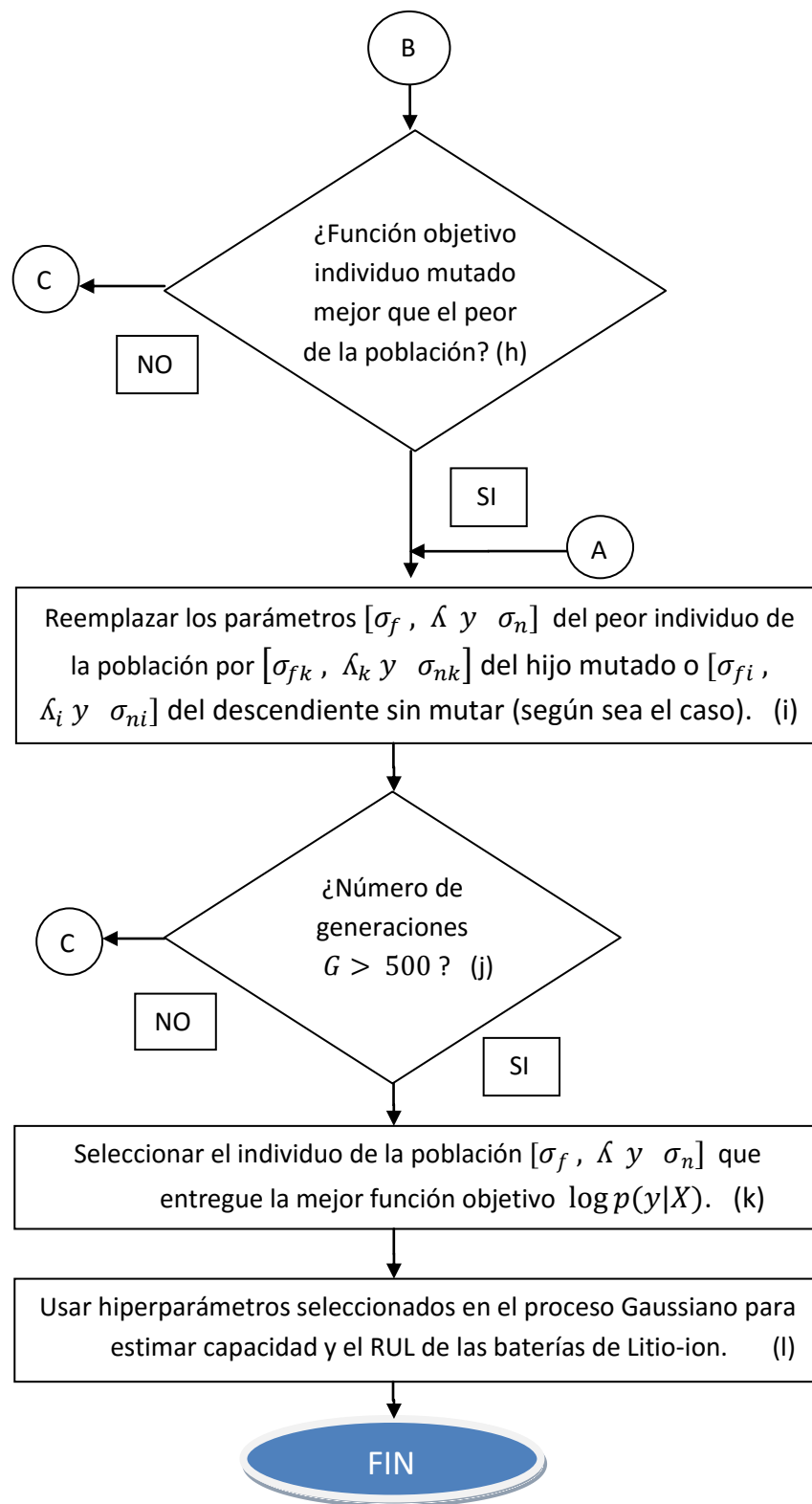


Figura 16. Diagrama de flujo proceso de estimación del RUL para baterías de Litio-ion.

Contextualización del diagrama de flujo de la figura 16:

Bloque (a): Generación de la población inicial con individuos de hiperparámetros σ_f , λ y σ_n . Ver sección 5.4.1 (Figura 13) y sección 4.3.2 en donde se explica el concepto de población inicial.

Bloque (b): Se evalúa la bondad de cada individuo de la población en relación con la función objetivo $\log p(y|X)$, para así determinar que tan bueno como solución del problema. Ver sección 5.4.1 (Figura 14), sección 4.3.3 para ver más información sobre la función objetivo y verificar su respectiva fórmula mediante la ecuación (11).

Bloque (c): Se escogen dos padres de los 500 individuos de la población para la generación de un nuevo descendiente. En la sección 4.3.4 se explica el proceso de selección de individuos mediante el método "selección por ruleta", se sugiere ver Figura 5.

Bloque (d): Se crea un nuevo descendiente mediante el proceso de recombinación heurística con hiperparámetros $[\sigma_{fi}, \lambda_i \text{ y } \sigma_{ni}]$. Se recomienda ver sección 4.3.5, donde se explica la metodología utilizada y se presenta un ejemplo numérico.

Bloque (e): Se efectúa el proceso de mutación no uniforme si se cumple cierto criterio de probabilidad., para este caso, valor de $L < 0.1$. Ver sección 5.4.1 (pág. 47).

Bloque (f): Se tiene un nuevo hijo con cierta modificación después del proceso de recombinación heurística con hiperparámetros $[\sigma_{fk}, \lambda_k \text{ y } \sigma_{nk}]$. Ver sección 4.3.6, donde se explica el método de operación de la mutación, además se muestra un ejemplo numérico en la figura 9.

Bloque (g): Se evalúa bondad del nuevo individuo mutado. Ver sección 5.4.1 (Figura 14), sección 4.3.3 para ver más información sobre la función objetivo y verificar su respectiva fórmula mediante la ecuación (11).

Bloque (h): Se realiza una comparación entre la función objetivo del hijo mutado y el que tenga peor rendimiento en la población actual. Ver sección 5.4.1 (pág. 48).

Bloque (i): Según sea el caso, se efectúa el reemplazo del nuevo descendiente, bien sea mutado o sin el efecto de la mutación. Ver figura 15 en el cual se presenta un esquemático del reemplazo.

Bloque (j): Básicamente en esta instancia se indaga si la cantidad de iteraciones o generaciones propuestas ya fue alcanzada. Ver sección 5.4.1 (pág. 44).

Bloque (k): Puesto que ya terminó la cantidad de iteraciones propuestas, se seleccionan los hiperparámetros $[\sigma_f, \lambda \text{ y } \sigma_n]$ que minimizan la función objetivo de la ecuación (11).

Bloque (l): Se efectúa la estimación del RUL para un nuevo ciclo de baterías, con ayuda de los hiperparámetros estimados en el bloque (k), y mediante las ecuaciones (6), (9) y (10). Refiérase a la sección 4.2 para verificar el concepto de la estimación.

Con el fin de condensar la metodología propuesta para la estimación del tiempo de vida útil restante de las baterías de Litio-ion, se propone verificar el diagrama de bloques de la figura 17, en la cual se muestra de una manera general, las entradas y las salidas que poseen los procesos más relevantes del método de optimización.

El bloque "Base de datos" corresponde a la información contenida en [26] para todas las baterías de Litio-ion, de allí se dividen dichos datos en grupo de entrenamiento y grupo de prueba. La etapa de entrenamiento, tiene como fin estimar los hiperparámetros $[\sigma_f, \lambda \text{ y } \sigma_n]$ óptimos de la función de covarianza para que finalmente, en el bloque "etapa de validación" se calcule la función media y de covarianza para la capacidad de las baterías de Litio-ion del grupo de prueba seleccionado inicialmente.

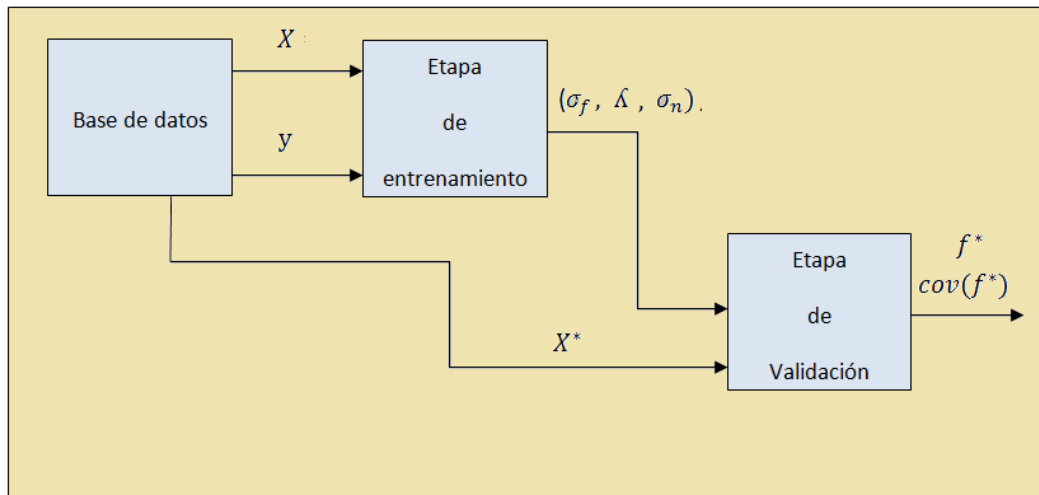


Figura 17. Diagrama de bloques general proceso de estimación del RUL de las baterías Litio-ion.

Las variables y parámetros que se utilizan en alguna instancia del modelo para poder efectuar la estimación del tiempo de vida útil de las baterías de Litio-ion, se presentan a continuación:

Variables:

X = Conjunto del Training Set sin datos de capacidad de la batería en los ciclos.

X^* = Test Set sin datos de capacidad de la batería en los ciclos.

y = Datos de la capacidad de la batería para el Training Set.

f^* = Datos estimados de capacidad de baterías del Test Set.

Parámetros:

σ_f = Factor de escala aplicada a la función Kernel seleccionada (ver sección 4.2)

λ = Longitud de escala de la función Kernel seleccionada (ver sección 4.2)

σ_n^2 = Factor de escala que representa la cantidad de ruido existente en " X " y en " y "

n = Número de ciclos de descarga de la batería (ver sección 5.2).

G_{Max} = Número máximo de generaciones del algoritmo genético de valor real.

G = Número de generaciones actuales para el algoritmo genético de valor real.

b = Parámetro de forma para la operación de mutación (ver sección 4.3.6).

r = Dato aleatorio uniformemente distribuido entre 0 y 1.

Limite Superior = Valor máximo que puede alcanzar un hiperparámetro.

Limite Inferior = Valor mínimo que puede tener un hiperparámetro.

Definiciones:

$K(X_i, X_j)$ = Función de covarianza seleccionada (ver sección 4.2, ecuación (6)).

$\log p(y|X)$ = Función de costo u objetivo \rightarrow (ver sección 4.2, ecuación (11)).

6. RESULTADOS

En esta sección, se muestran y se discuten los resultados de la metodología propuesta para la estimación del tiempo de vida útil de baterías de Litio-ion. Como primer elemento, se expone la cantidad de ciclos existentes para cada uno de los dispositivos que componen la base de datos usada [26], con la necesidad de extraer únicamente las muestras de descarga de las baterías para efectuar el proceso de regresión. Como segundo ítem, se exponen los resultados obtenidos mediante la metodología implementada. Finalmente, se comparan dichos resultados con los existentes en la literatura, para evaluar rendimiento y eficiencia en la operación.

6.1 FILTRACIÓN DE CICLOS

Considerando que la metodología propuesta fundamenta su operación en los ciclos de descarga de las baterías, se hizo relevante efectuar una clasificación exhaustiva a cada uno de los elementos que componen la base de datos [26], con el fin exclusivo de extraer cada uno de estos ciclos, ya que a partir de allí se genera la matriz de entrada y salida del proceso Gaussiano (Ver Sección 5.2).

Los resultados de este filtro inicial, se pueden verificar en la tabla 1.

Número Batería	Ciclos de carga	Ciclos de Descarga	Ciclo de impedancia	TOTAL DE CICLOS
Batería 5	170	168	278	616
Batería 6	170	168	278	616
Batería 7	170	168	278	616
Batería 18	134	132	53	319
Batería 25	31	28	21	80
Batería 26	31	28	21	80
Batería 27	31	28	21	80
Batería 28	31	28	21	80
Batería 29	40	40	17	97
Batería 30	40	40	17	97
Batería 31	40	40	17	97
Batería 32	40	40	17	97
Batería 33	197	197	92	486
Batería 34	197	197	92	486
Batería 36	197	197	92	486
Batería 38	47	47	28	122
Batería 39	47	47	28	122

Batería 40	47	47	28	122
Batería 41	67	67	29	163
Batería 42	113	112	50	275
Batería 43	113	112	50	275
Batería 44	113	112	50	275
Batería 45	72	72	40	184
Batería 46	72	72	40	184
Batería 47	72	72	40	184
Batería 48	72	72	40	184
Batería 49	25	25	12	62
Batería 50	25	25	12	62
Batería 51	25	25	12	62
Batería 52	25	25	12	62
Batería 53	55	55	26	136
Batería 54	102	103	48	253
Batería 55	102	102	48	252
Batería 56	102	102	48	252

Tabla 1. Clasificación de ciclos para la base de datos del repositorio de la NASA para baterías de Litio-Ion.

Luego de tener la información anterior, se generó una subdivisión a cada uno de los ciclos de descarga de cada batería, con el fin de generar los datos de entrenamiento y de prueba para el modelo de regresión de acuerdo a lo expuesto en la sección 5.4 del presente documento. En la tabla 2, se muestra como se distribuyeron estos datos para cada una de las baterías estudiadas.

Número Batería	Total Ciclos de Descarga	Datos de entrenamiento	Datos de prueba
Batería 5	168	118	50
Batería 6	168	118	50
Batería 7	168	118	50
Batería 18	132	92	40
Batería 25	28	20	8
Batería 26	28	20	8
Batería 27	28	20	8
Batería 28	28	20	8
Batería 29	40	28	12
Batería 30	40	28	12
Batería 31	40	28	12
Batería 32	40	28	12
Batería 33	197	138	59

Batería 34	197	138	59
Batería 36	197	138	59
Batería 38	47	33	14
Batería 39	47	33	14
Batería 40	47	33	14
Batería 41	67	47	20
Batería 42	112	78	34
Batería 43	112	78	34
Batería 44	112	78	34
Batería 45	72	50	22
Batería 46	72	50	22
Batería 47	72	50	22
Batería 48	72	50	22
Batería 49	25	18	7
Batería 51	25	18	7
Batería 53	55	39	16
Batería 54	103	72	31
Batería 55	102	71	31
Batería 56	102	71	31

Tabla 2. Subdivisión ciclos de descarga baterías de Litio-Ion para datos de entrenamiento y prueba.

6.2 REGRESIÓN CON PROCESO GAUSSIANO Y ALGORITMO GENÉTICO DE VALOR REAL

6.2.1 SELECCIÓN PARÁMETROS DEL ALGORITMO GENÉTICO

Como se puede verificar en la sección 4.2, el proceso Gaussiano usado para efectuar una regresión posee tres hiperparámetros, dos de ellos asociados al Kernel (σ_f , λ) que en este caso es una función exponencial cuadrada, y el tercero de ellos (σ_N) que está ligado al ruido de los datos de entrada para la creación del modelo. La optimización de estos hiperparámetros se realizó mediante la utilización de un algoritmo genético de valor real como el mostrado en el inciso 4.3 del presente trabajo. Es relevante tener en cuenta, que al correr repetidamente este algoritmo de optimización, los resultados (valores de hiperparámetros) pueden variar un poco entre sí, ya que este tipo de técnicas heurísticas posee diversas variables como el tamaño de la población y el número de generaciones, que influyen en las magnitudes finales de los hiperparámetros optimizados (ver sección 4.3). Por lo anterior, se decidió realizar una serie de corridas con distintos valores de tamaño de población y cantidad de generaciones para el algoritmo genético, todo con el fin de determinar para cual pareja de parámetros se obtenía una menor desviación para

los valores de los hiperparámetros (σ_f , λ , σ_N). Se probaron valores para la cantidad de generaciones como (100, 500 y 1000), además de magnitudes para el tamaño de la población de (20, 100, 500). En la tabla 3, 4 y 5 se muestran los resultados de las corridas obtenidos para la batería de Litio-Ion 54 con una población fija de 20 individuos y diversos tamaños de población, cabe anotar que este dispositivo fue seleccionado para la prueba, puesto que cuenta con una cantidad de ciclos de descarga media, permitiendo llevar las conclusiones obtenidas tanto para baterías con un número de ciclos de descarga mayor y menor.

Batería 54				
# Corrida	Tamaño Población - Número Generaciones	σ_f	λ	σ_N
1	20 - 100	7,11690762	-1,77450486	-0,91822789
2	20 - 100	6,67992317	1,25093659	-0,80885401
3	20 - 100	6,64021181	-2,375748	-1,44265448
4	20 - 100	7,47390642	3,91990401	-6,68481301
5	20 - 100	8,0168284	2,9945508	-4,07854627
6	20 - 100	-3,97081613	-1,75323017	-0,46792352
7	20 - 100	-2,60666515	-0,04183361	-2,22823314
8	20 - 100	2,00624562	0,14287988	-1,68934065
9	20 - 100	6,97361153	3,39000915	-0,24968577
10	20 - 100	0,71925861	0,79967313	0,74977723
Desviación estándar hiperparámetros		4,50608249	2,25153738	2,15541593

Tabla 3. Valores hiperparámetros para tamaño población = 20 y con un máximo de 100 generaciones.

Batería 54				
# Corrida	Tamaño Población - Número Generaciones	σ_f	λ	σ_N
1	20 - 500	7,99743945	1,92185314	-4,26095406
2	20 - 500	9,59939064	4,99968006	-0,84899787
3	20 - 500	7,32288861	3,37929266	-4,53754985
4	20 - 500	8,40657799	4,7432899	-6,34977818
5	20 - 500	9,7643716	8,28543475	-4,32681657
6	20 - 500	6,68982499	5,00681219	-8,06199282
7	20 - 500	7,96767884	5,64929447	-3,50552631
8	20 - 500	9,40949459	6,62221032	-4,95952754
9	20 - 500	7,48551572	4,05160214	-6,65825649
10	20 - 500	9,80395549	-1,7016833	-0,19798945
Desviación estándar hiperparámetros		1,13237086	2,72575368	2,44327329

Tabla 4. Valores hiperparámetros para tamaño población = 20 y con un máximo de 500 generaciones.

Batería 54				
# Corrida	Tamaño Población - Número Generaciones	σ_f	$\hat{\lambda}$	σ_N
1	20 - 1000	6,26484434	2,38198033	-6,53032741
2	20 - 1000	9,59766941	6,90399701	-1,33386172
3	20 - 1000	9,84928791	6,16103638	-4,92917771
4	20 - 1000	9,68099162	8,77481967	-3,64272985
5	20 - 1000	5,73804118	-0,56199454	-3,38709357
6	20 - 1000	9,66408595	8,47726541	-6,0624566
7	20 - 1000	7,57833306	1,90063247	-3,92803517
8	20 - 1000	9,70722191	8,44466552	-0,83303279
9	20 - 1000	8,37753621	7,15436628	-4,36030208
10	20 - 1000	2,62208033	0,14824792	-2,78865208
Desviación estándar hiperparámetro		2,39830707	3,6324407	1,83416281

Tabla 5. Valores hiperparámetros para tamaño población = 20 y con un máximo de 1000 generaciones.

En las tablas anteriores, no es posible observar un cambio significativo en la desviación estándar de los hiperparámetros, mediante la modificación de la cantidad de generaciones soportadas por el algoritmo genético. En la tabla 6, se puede verificar que la desviación estándar de los hiperparámetros tanto del Kernel como de la función de verosimilitud van descendiendo a medida que se aumenta el tamaño de la población, haciéndose cada vez más estable. Lo anterior se comenta con el propósito de afirmar que al efectuar diferentes corridas para una población inicial de 500 individuos, se encuentran unos mejores resultados que al hacerlo con poblaciones con tamaños menores. Es por lo anterior, y teniendo en cuenta la poca variación efectiva que produce aumentar o disminuir la cantidad de generaciones (ver Tabla 6), que se decidió trabajar la metodología propuesta con una población inicial de 500 individuos y con un número máximo de generaciones equivalente a 500.

Desviaciones Estándar hiperparámetros (10 corridas)			
Tamaño Población - Número Generaciones	σ_f	λ	σ_N
20 - 100	4,506082489	2,251537381	2,155415925
20 - 500	1,132370856	2,725753684	2,443273288
20 - 1000	2,398307068	3,6324407	1,83416281
100 - 100	1,32413962	1,903981949	1,466563442
100 - 500	2,229509081	3,39061346	0,806853263
100 - 1000	2,579238945	3,493538614	0,926429698
500 - 100	0,966736172	2,285781078	0,495468926
500 - 500	1,252601479	1,329907569	0,497451683
500 - 1000	1,173456762	1,859467678	0,632083805

Tabla 6. Condensado de las desviaciones estándar de los hiperparámetros en las distintas corridas para la batería 54.

Vale resaltar, que aunque el tamaño de la población podría incrementarse más para obtener posiblemente una pequeña mejora en el cálculo del RUL, computacionalmente empieza a ser muy costoso efectuar todas las operaciones que implica resolver la optimización de los hiperparámetros mediante este tipo de técnicas.

6.2.2 APLICACIÓN DEL MODELO PROPUESTO

Al definirse dos de los parámetros más relevante para el rendimiento del algoritmo genético, como lo son el tamaño de población y la cantidad de generaciones, se propone ahora la asignación de los valores de probabilidad para la realización de los procedimientos de recombinación y de mutación. Como se comentó en la sección 5.4.1, Para el desarrollo del presente proyecto, se seleccionó un valor de 1 para la recombinación, puesto que se asume que siempre que existan dos padres existirá un hijo; por su parte, la operación de mutación tiene una tasa equivalente a 0,1, ya que haciendo analogía con la naturaleza, se espera que no sea muy frecuente encontrar descendientes con alguna alteración.

Ya con la parametrización total del algoritmo genético, se puede ejecutar en su totalidad la metodología propuesta, ya que los hiperparámetros del proceso Gaussiano pueden ser estimados mediante este procedimiento.

Para verificar el buen funcionamiento de la técnica, se decide efectuar una prueba inicial con la batería de Litio-Ion 54 para observar su rendimiento con los 31 ciclos de test que se seleccionaron. En la figura 18 es posible chequear este resultado, allí el eje de abscisas hace referencia al número de ciclo de descarga correspondiente en el test set, mientras en las ordenadas se encuentra la capacidad en Amperes-hora de la batería de Litio-Ion de cada ciclo en específico.

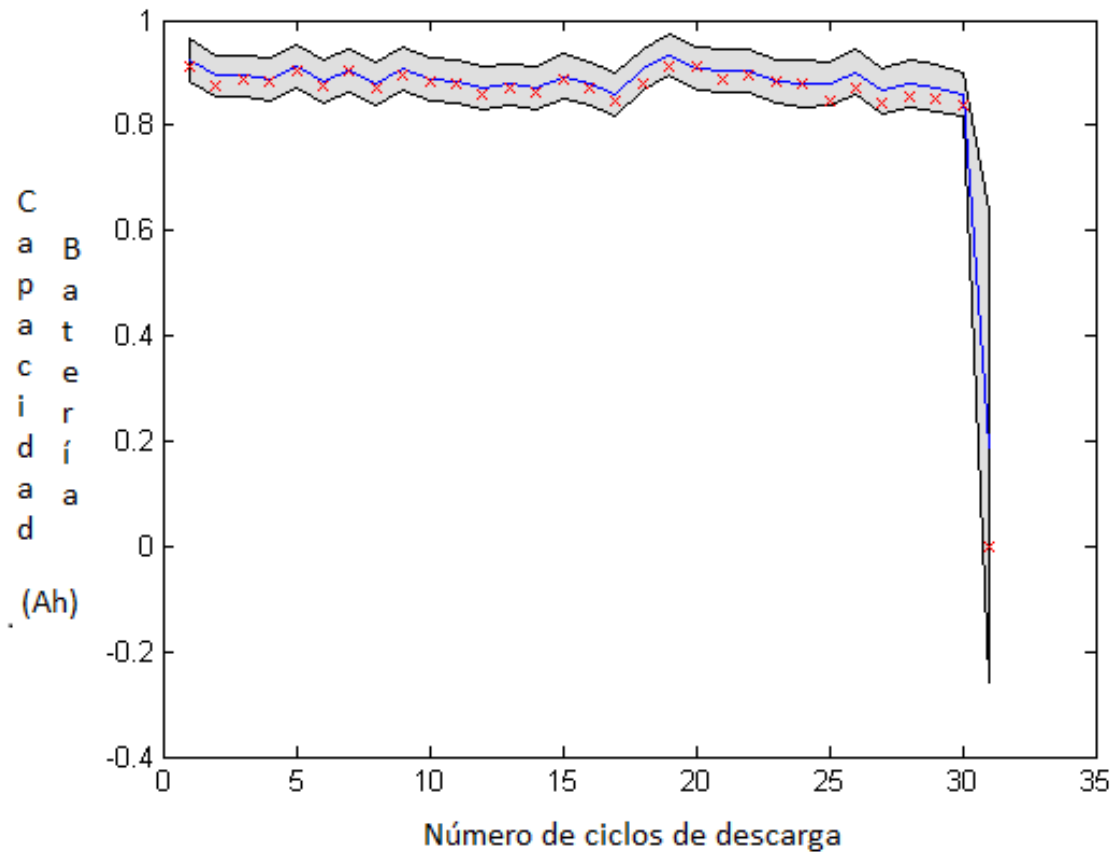


Figura 18. Método de regresión para la batería 54 de la base de datos [26].

En la figura anterior, es válido rescatar las siguientes características o aspectos:

- Las cruces de color rojo corresponden a las magnitudes de capacidad de la batería para cada uno de los datos del test set.
- La línea azul hace referencia al valor de la media de la regresión hecha mediante el proceso Gaussiano + Algoritmo genético de valor real.
- El área sombreada de color gris representa la varianza calculada mediante el proceso de regresión.

Es relevante comentar, que entre más cerca se encuentre la línea azul de cada una de los cruces rojos, se puede decir que el rendimiento del proceso de regresión fue bueno. Cabe decir también, que se espera que la varianza sea pequeña indicando que la operación de estimación fue exitosa y cumple con las expectativas planteadas inicialmente. Para darle un valor numérico a la estimación realizada, se evalúa el rendimiento de la regresión por medio del índice RSE (*Root Square Error*) [9], la cual expresa una desviación o diferencia entre el estimador (datos del modelo de regresión) y la variable estimada que en este caso es la capacidad de la batería. El RSE está dado por la siguiente expresión [9]:

$$RSE = \sqrt{\sum_{i=1}^N d_i^2} \quad (22)$$

donde:

$d_i = \text{Valor dado por el modelo}_i - \text{Valor real}_i$

$N = \text{Cantidad de datos evaluados (Longitud del test set)}$

Para el caso de la figura 18, se tiene un RSE igual a $6,3 \times 10^{-3}$, el cual es una magnitud baja indicando que se tiene un muy buen resultado. Esta premisa es reafirmada debido al hecho de que al correr el algoritmo por medio del Toolbox de procesos Gaussianos de Matlab para esta batería 54, se obtiene un $RSE = 6.8627$. Lo anterior muestra una mejora significativa del algoritmo propuesto con respecto al ejecutado en el ToolBox.

En la tabla 7, se muestran los valores arrojados para los hiperparámetros efectuando la metodología propuesta y el ToolBox de procesos Gaussianos. Se puede observar claramente que ambos difieren mucho, entendiéndose el motivo de resultados tan diferentes al momento de la predicción (ver figura 19), ya que en esta última figura los resultados son muy pobres, prediciendo siempre un valor de cero.

	HIPERPARÁMETROS		
PROCEDIMIENTO	σ_f	$\hat{\lambda}$	σ_N
Metodología propuesta	6.5143	-1.0543	-3.8522
ToolBox GPR	3.22×10^{-12}	0.0922	-2.3017

Tabla 7. Valor hiperparámetros para cada procedimiento evaluado

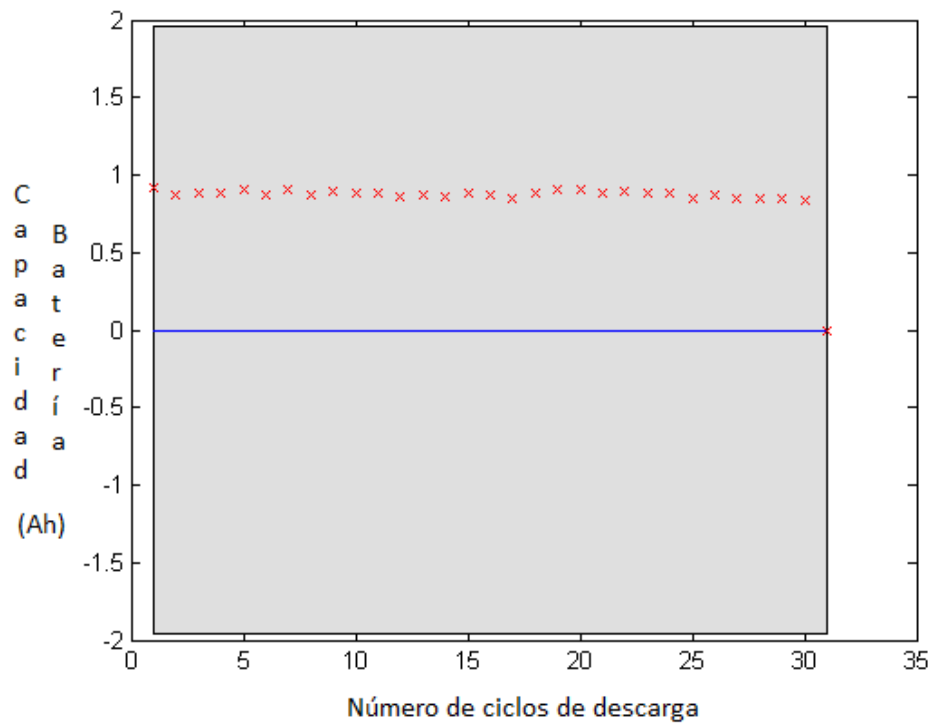


Figura 19. Ejecución del Toolbox de Matlab para la regresión de la batería 54 de la base de datos [26].

6.3 COMPARACIÓN RESULTADOS TOOLBOX vs METODOLOGÍA PROPUESTA

Al revisar previamente la figura 18 y la figura 19, se observa que la aplicación de un proceso Gaussiano para regresión en conjunto con un algoritmo genético de valor real, representa una estimación excelente para el caso puntual de la batería de Litio-Ion 54; pero, ¿Qué pasa si el tamaño del test set aumenta o disminuye?, ¿Sigue teniendo la metodología propuesta un mejor rendimiento?.

Con el fin de dar resolución al planteamiento anterior, se decidió condensar los resultados de todas las baterías de la base de datos en la tabla 8. Allí, se registran los diferentes valores del RSE para cada operación permitiendo efectuar comparaciones entre ambos modelos, además de las magnitudes promedio de las varianzas para los pronósticos del RUL de la batería.

VALOR RSE Y VARIANZA PARA TÉCNICAS DE REGRESIÓN USADAS				
Número Batería	ToolBox GPR Matlab		Metodología Propuesta	
	RSE	σ^2	RSE	σ^2
Batería 5	0.0023	283.4340	0.0045	2.87×10^{-7}
Batería 6	0.0312	1.58×10^{-5}	0.0533	1.67×10^{-4}
Batería 7	0.0309	1.23×10^{-5}	0.0188	1.12×10^{-4}
Batería 18	0.0167	5.21×10^{-6}	0.0195	1.95×10^{-5}
Batería 25	0.1033	3.05×10^{-4}	0.0096	7.11×10^{-6}
Batería 26	5.0384	3.1553	0.0194	5.87×10^{-5}
Batería 27	0.0177	3.05×10^{-5}	0.0176	1.30×10^{-4}
Batería 28	4.9299	3.1719	0.0024	2.58×10^{-6}
Batería 29	0.0288	3.88×10^{-5}	0.0037	5.12×10^{-5}
Batería 30	0.0231	2.42×10^{-5}	0.0218	1.87×10^{-4}
Batería 31	0.0210	2.94×10^{-5}	0.0209	4.81×10^{-5}
Batería 32	2.29×10^{-4}	6.33×10^{-4}	2.90×10^{-4}	8.49×10^{-4}
Batería 33	0.9781	0.0016	0.7977	0.0015
Batería 34	0.0611	4.58×10^{-4}	0.0930	1.94×10^{-4}
Batería 36	0.0035	0.2397	0.0105	0.0027
Batería 38	6.4538	12.4077	1.1766	11.912
Batería 39	6.1471	2.0438	1.3461	0.0252
Batería 40	0.9438	0.0244	0.7512	0.0743
Batería 41	0.1034	5.98×10^{-4}	0.2432	0.0034
Batería 42	7.1787	0.6401	9.6575	3.4253
Batería 43	7.1038	0.4176	6.1598	2.8651
Batería 44	6.8627	1.2125	6.8602	0.3638
Batería 45	2.9102	0.5433	0.5002	8.91×10^{-4}
Batería 46	5.2383	1.7431	0.7886	5.50×10^{-4}
Batería 47	5.2099	1.6541	0.8878	7.75×10^{-5}
Batería 48	5.5076	1.7373	1.0247	2.35×10^{-4}
Batería 49	1.9366	1.1937	0.0922	9.25×10^{-4}
Batería 51	1.9274	1.0755	0.0673	0.0024
Batería 53	4.1619	1.1118	0.1695	0.0082
Batería 54	4.8061	0.9631	0.4437	0.0091
Batería 55	0.0498	4.90×10^{-5}	0.0462	1.08×10^{-4}
Batería 56	6.3474	1.4290	0.0099	1.90×10^{-5}

Tabla 8. Resultados de comparación entre el ToolBox GPR Matlab y la metodología propuesta.

Como se puede observar en la tabla 8, los valores resaltados en amarillo representan aquellos resultados que tuvieron un mejor rendimiento para una batería en específico. Como apunte general, se concluye que la metodología propuesta tiene siempre un rendimiento superlativo para todas las series de baterías analizadas en la base de datos [26]. Por ejemplo, para el caso específico de la batería 18, el ToolBox GPR de Matlab supera por algunas centésimas al método de GPR + Algoritmo genético, obteniendo ambos unos resultados excelentes. Pero en otros casos, como el de la batería 49, donde la metodología propuesta posee un rendimiento considerablemente mejor que el del ToolBox de Matlab. En la figura 20 se representa mediante un diagrama de torta el rendimiento de las dos metodologías comparadas tanto como el RSE como para σ^2 . Por ejemplo, con respecto al RSE, en el 75% de los casos analizados (24 de 32), la metodología propuesta obtiene un mejor resultado, así como para el ítem de la varianza en la estimación hay un mejor rendimiento en el 56,3% de las veces (18 de 32).

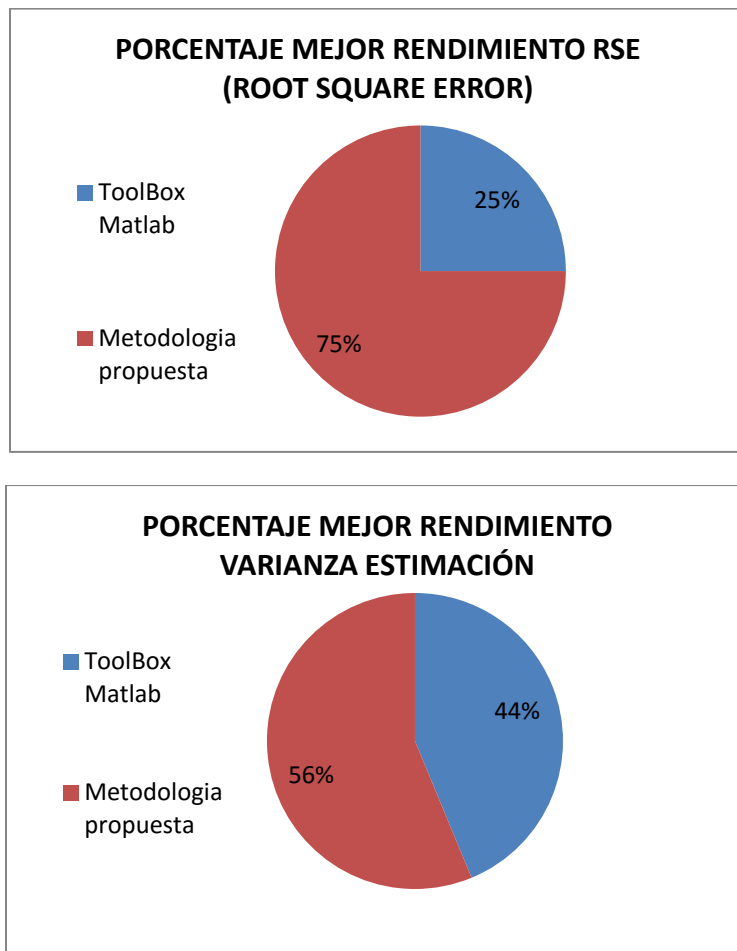


Figura 20. Rendimiento porcentual comparación metodologías.

Ahora bien, con el fin de analizar algunos casos específicos, en la figura 21 se expone las gráficas de regresión para los datos test set de la batería 5, en donde se tienen raíces cuadradas del error de predicción (RSE) muy similares, pero el modelo propuesto expone una varianza mucho más estrecha que la del ToolBox. Lo anterior quiere decir, que aunque ambos modelos poseen errores de predicción bajos y similares, la metodología propuesta representa una opción mucho más confiable, ya que su intervalo de incertidumbre es menor.

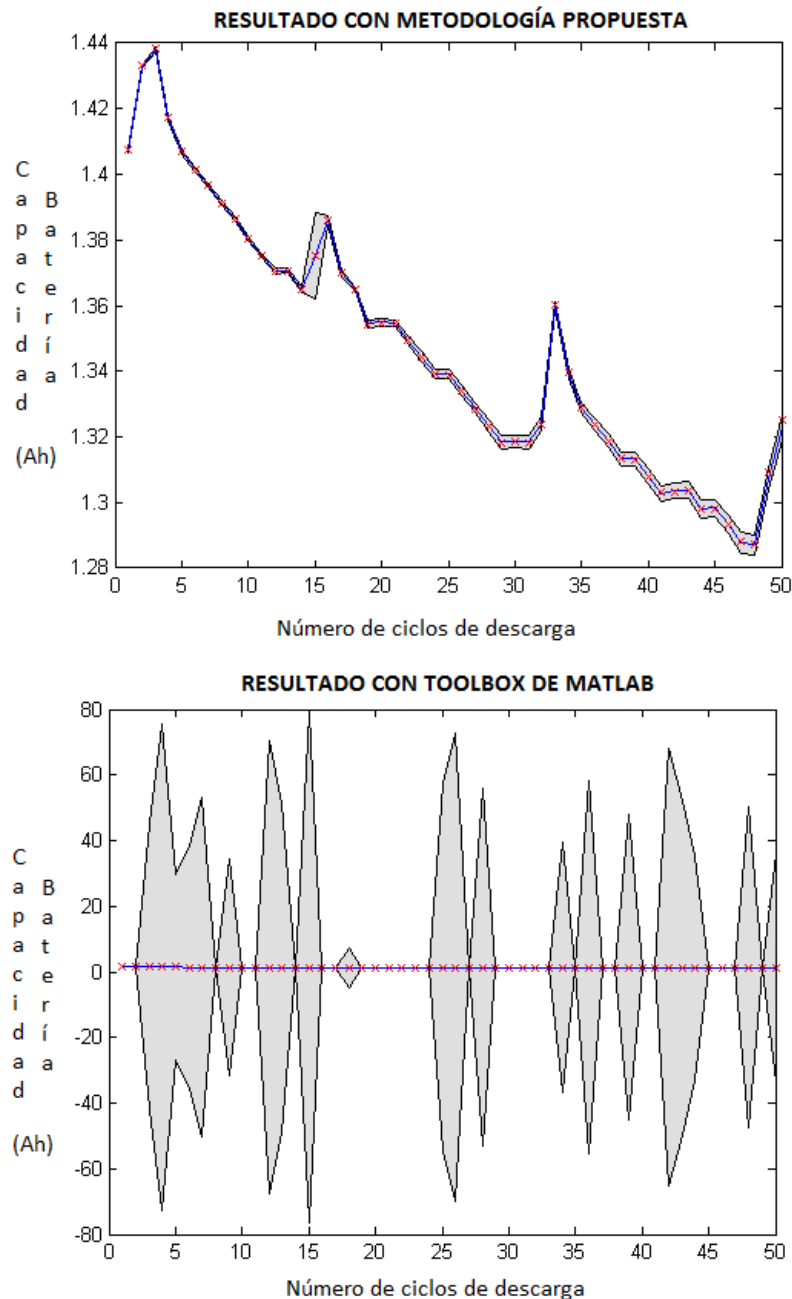


Figura 21. Comparación resultados metodología propuesta y Toolbox GPR para batería 5.

La figura 22 también representa los buenos resultados que se obtienen al calcular los hiperparámetros del Kernel de un proceso Gaussiano mediante un algoritmo genético de valor real, puesto que mediante el ToolBox de Matlab se estima para todo el test set una señal constante de valor cero, mientras que, con la metodología propuesta en el presente proyecto, se alcanzan a reconocer todos los cambios adquiridos por la señal en los datos de entrenamiento, arrojando estimaciones adecuadas.

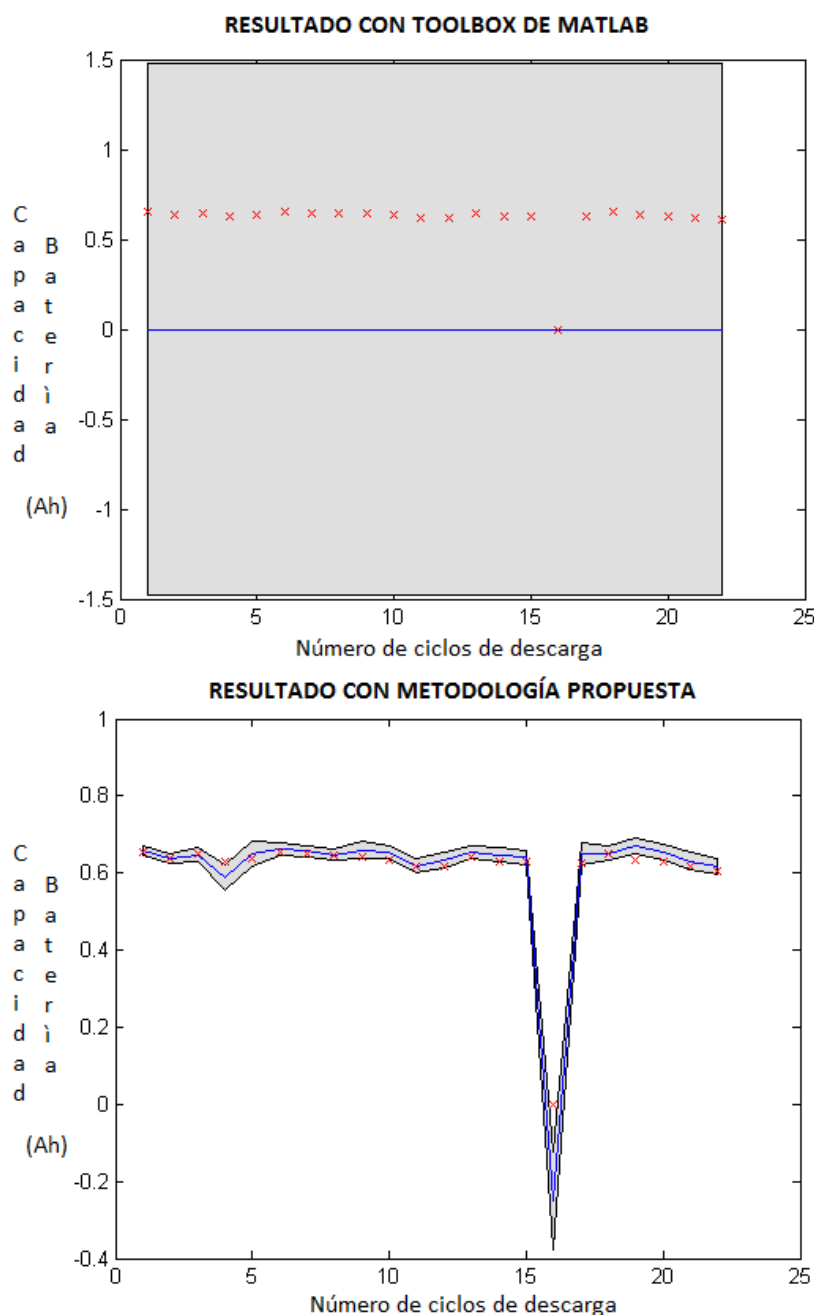
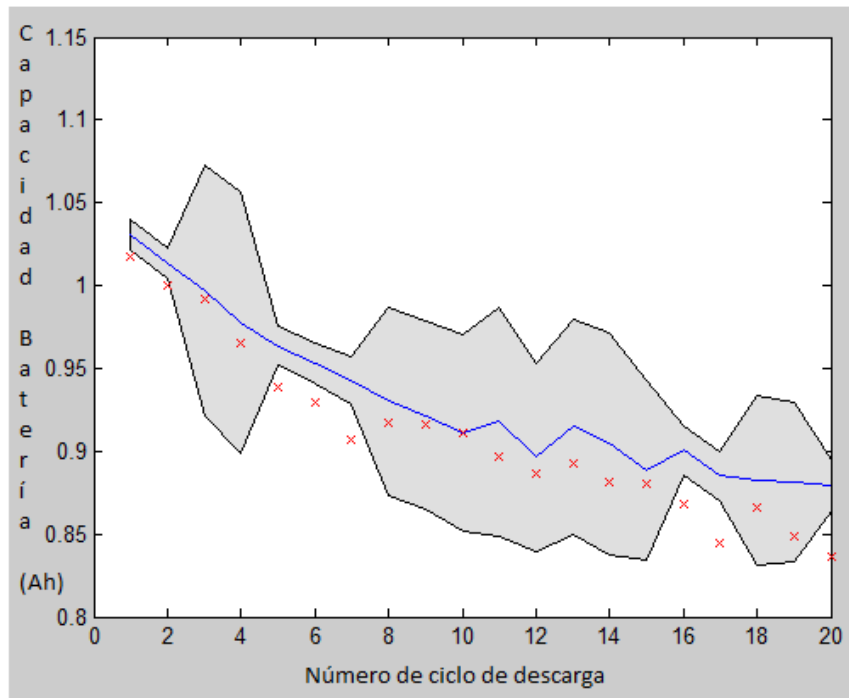


Figura 22. Comparación resultados metodología propuesta frente al ToolBox GPR para batería 45.

Es válido analizar también una batería de la base de datos en la cual el ToolBox de Matlab presente mejores índices (RSE , σ^2) que la metodología propuesta para verificar si existen diferencias tan evidentes como en las dos figuras anteriores.

RESULTADO CON TOOLBOX MATLAB



RESULTADO CON METODOLOGÍA PROPUESTA

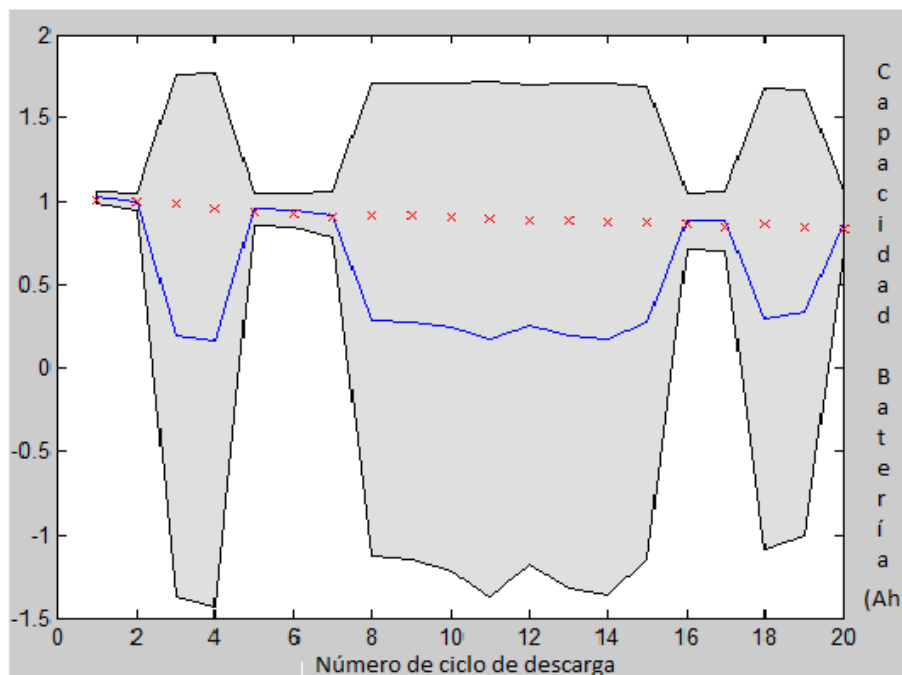


Figura 23. Comparación resultados metodología propuesta frente al ToolBox GPR para batería 41.

Al analizar detalladamente la figura anterior, se constata la información expuesta en la tabla 8, en donde el ToolBox de Matlab desarrolla un error RSE menor y una varianza σ^2 más pequeña que la generada por la metodología propuesta. La cuestión a destacar también, es que aunque los índices son un poco mejores, existen algunos datos del test set de la batería 41, que no están siendo abarcados por la varianza total que entrega el ToolBox de Matlab (Observar que datos 5,6,7,16 Y 17 se encuentran fuera del intervalo de confianza. Por su parte, la metodología propuesta (GPR + Algoritmo genético de valor real) ubica a cada uno de los datos de entrenamiento dentro del intervalo de confianza entregado.

6.4 COMPARACIÓN RESULTADOS LITERATURA

Es interesante contrastar el rendimiento de la metodología propuesta con respecto a otro tipo de técnicas que se conocen han tenido éxito para este prototipo de casos en específico. Mayoritariamente, se usa el concepto de "*función score*" para evaluar el rendimiento del modelo propuesto. Este indicador se usa para cuantificar que tan bueno o malo es un modelo de acuerdo a unos parámetros seleccionados que ingresan a una función matemática. Por ejemplo, En la conferencia "IEEE Prognostics and Health Management (PHM)" realizada en el 2008, [9, 51] usa el concepto de "*función score*" para evaluar el rendimiento de su modelo mediante una red neuronal y un filtro de Kalman. Por lo anterior, se decidió acoger este concepto para realizar la comparación entre el modelo descrito en el presente documento y el expuesto en [9], ya que este último obtuvo el mejor puntaje para la estimación del RUL de la base de datos [26] en la conferencia PHM del año 2008, la cual es utilizada también en el actual trabajo.

El function score seleccionado para la comparación de los modelos es el siguiente [9]:

$$S = \begin{cases} e^{\frac{-d_i}{13}} - 1, & d_i \leq 0 \\ e^{\frac{-d_i}{10}} - 1, & d_i \geq 0 \end{cases} \quad i = 1, \dots, N \quad (23)$$

donde:

$$d_i = \text{RUL Estimado}_i - \text{RUL Real}_i$$

$$N = \text{Cantidad de muestras estudiadas}$$

Cabe anotar, que en el artículo [9], se obtuvo un valor de $S = 5636$ para un total de 218 predicciones ($N = 218$). Ahora bien, en la tabla 9 se condensa la información entregada por [26] referente al porcentaje al cual se considera que cada batería ha terminado su ciclo de vida útil. Cabe anotar que en algunos de los dispositivos incluidos en la base de

datos, no se presenta información sobre este criterio, por lo cual no se tuvieron en cuenta para calcular el índice S hallado en [9]. Además de lo anterior, los datos de capacidad de ciertas baterías ya se encontraban por debajo del umbral admisible como útil, por lo cual tampoco fueron analizadas. En resumen, se consideran 15 baterías para efectuar el cálculo de la variable S descrita en la ecuación (23).

Número Batería	Capacidad nominal (Ah)	Criterio de Fin Vida útil (Ah)	Ciclo de Fin Vida útil Real	Ciclo de Fin Vida Estimado	d_i
Batería 5	2	1,4 (-30%)	125	125	0
Batería 6	2	1,4 (-30%)	109	109	0
Batería 7	2	1,4 (-30%)	N.A	N.A	N.A
Batería 18	2	1,4 (-30%)	97	97	0
Batería 25	2	N.A	N.A	N.A	N.A
Batería 26	2	N.A	N.A	N.A	N.A
Batería 27	2	N.A	N.A	N.A	N.A
Batería 28	2	N.A	N.A	N.A	N.A
Batería 29	2	N.A	N.A	N.A	N.A
Batería 30	2	N.A	N.A	N.A	N.A
Batería 31	2	N.A	N.A	N.A	N.A
Batería 32	2	N.A	N.A	N.A	N.A
Batería 33	2	1,6 (-20%)	53	53	0
Batería 34	2	1,6 (-20%)	23	23	0
Batería 36	2	1,6 (-20%)	186	186	0
Batería 38	2	1,6 (-20%)	46	39	-7
Batería 39	2	1,6 (-20%)	46	40	-6
Batería 40	2	1,6 (-20%)	46	40	-6
Batería 41	2	1,4 (-30%)	N.A	N.A	N.A
Batería 42	2	1,4 (-30%)	42	42	0
Batería 43	2	1,4 (-30%)	42	42	0
Batería 44	2	1,4 (-30%)	42	42	0
Batería 45	2	1,4 (-30%)	N.A	N.A	N.A
Batería 46	2	1,4 (-30%)	22	22	0
Batería 47	2	1,4 (-30%)	11	10	-1
Batería 48	2	1,4 (-30%)	16	16	0
Batería 49	2	N.A	N.A	N.A	N.A
Batería 51	2	N.A	N.A	N.A	N.A
Batería 53	2	1,4 (-30%)	N.A	N.A	N.A
Batería 54	2	1,4 (-30%)	N.A	N.A	N.A
Batería 55	2	1,4 (-30%)	N.A	N.A	N.A
Batería 56	2	1,4 (-30%)	N.A	N.A	N.A

Tabla 9. Vida útil baterías según información suministrada por [26].

La figura 24 y 25 exponen algunos casos de estimación del RUL mediante la metodología propuesta para las baterías 40 y 47 respectivamente. Es importante resaltar, que donde existió la mayor diferencia d_i fue con las baterías en las que se contaba con una cantidad menor de datos de entrenamiento (Baterías 38 , 39 y 40), siempre respetando la relación 70 - 30% entre el training y el test set respectivamente. Además, estas baterías presentan una particularidad y es el hecho de que poseen cambios muy bruscos en su capacidad (Ah) a medida que se van efectuando los procesos de descarga.

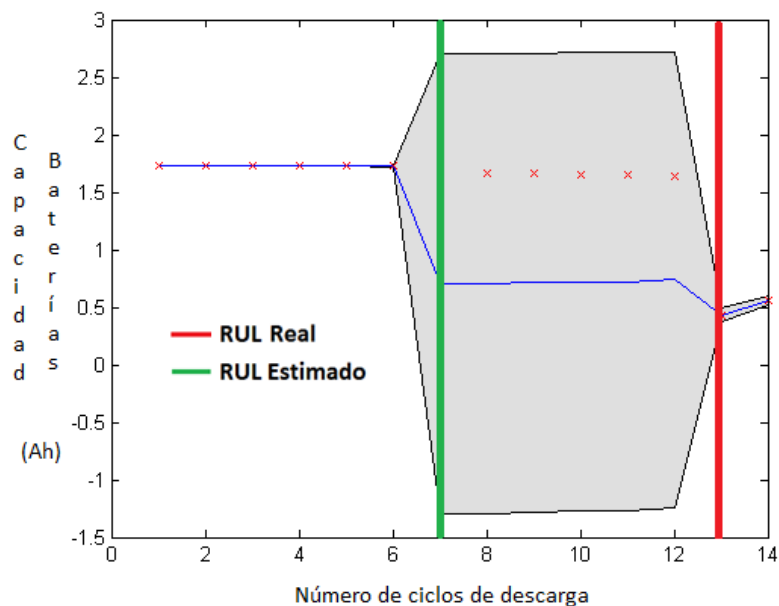


Figura 24. Verificación RUL Estimado vs RUL Real para la batería 40

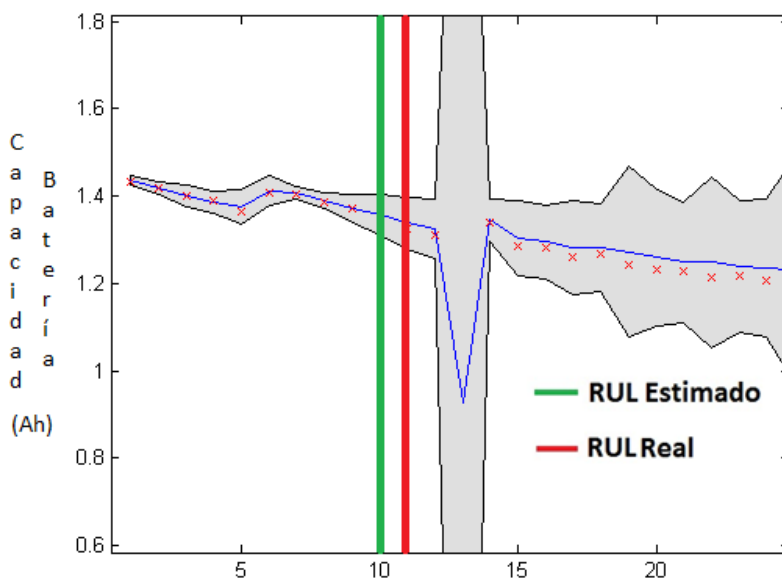


Figura 25. Verificación RUL Estimado vs RUL Real para la batería 47

Finalmente al calcular el índice S con los datos contenidos en la tabla 9, se obtiene que $S = 1.9664$, obteniéndose que cada una de las 15 baterías aporta al índice S un promedio de 0.1311. Ahora bien, si se efectúa este procedimiento para la variable S hallada por Wang en [51], se tiene que cada una de las 218 baterías aportaría un valor promedio de 25.853, indicando claramente que la metodología propuesta supera claramente el método de Wang , y que aunque no se han analizado las 218 muestras, su promedio de contribución para cada batería es mucho más bajo, soportando aun más los resultados de la tabla 8, donde se muestran RSE y varianzas relativamente bajas para la predicción.

7. CONCLUSIONES

En este proyecto, se desarrolló una metodología que permite estimar el tiempo de vida útil restante de una batería, mediante la aplicación de un Proceso Gaussiano cuyos hiperparámetros se encuentran seleccionados por el desarrollo de un algoritmo genético de valor real. Se muestra que este método mejora los resultados obtenidos en la predicción, ya que contar con herramientas de tipo heurísticas permite abarcar por todo una mayor parte del espacio de soluciones totales.

Se implementó una caracterización de la matriz de entrada/salida del Proceso Gaussiano basada principalmente en las medidas de distribución de las series de tiempo de voltaje y temperatura para cada uno de los ciclos de descarga de las baterías de Litio-ion, dando una robustez mayor a otro tipo de modelos de la literatura que tienen en cuenta únicamente los ciclos restantes de descarga.

Se efectuaron diversas pruebas para determinar con qué valores de tamaño de población inicial y número de generaciones se iba a trabajar para el problema de estimación del RUL de las baterías. Según los resultados de este procedimiento, se detecta que aumentar el tamaño de la población juega un papel fundamental para la obtención de resultados menos variantes en cada corrida. Se establecen además, las magnitudes para la tasa de recombinación y mutación del algoritmo genético.

Finalmente, se ejecuta en su totalidad la metodología propuesta notándose una significativa mejora en la estimación del RUL cuando se tiene en cuenta un proceso de optimización basado en un algoritmo genético de valor real y no en técnicas basadas en gradiente. Se compara también el rendimiento del método propuesto frente a otros que se conoce fueron exitosos en el "IEEE Prognostics and Health Management".

En términos generales, se puede establecer que el desarrollar un ejercicio de regresión mediante un Proceso Gaussiano en conjunto con un algoritmo genético de valor real entrega resultados superlativos, pudiéndose por tanto, transversalizar este hecho y llevarlo a distintas aplicaciones y áreas de la ciencia.

8. TRABAJO FUTURO

Para este trabajo, se implementó una regresión por medio de un Proceso Gaussiano para la base de datos de baterías de Litio-Ion ubicadas en el repositorio de la NASA. En este, se usa una función media única en conjunto con una función Kernel general. Un trabajo futuro puede estar orientado a realizar un desarrollo de Procesos Gaussianos integrados, donde tanto la función media como la función de covarianza sean seleccionadas mediante una combinación de distintas funciones ya existentes, permitiendo posiblemente una mejora en los resultados descritos.

Aunque se ha obtenido una mejora substancial mediante la metodología propuesta, este método tiene aún un gran potencial para seguir mejorando, ya que se poseen diversos factores o variables que podrían ser modificadas para la obtención de un rendimiento más alto. Se propone por ende, definir una caracterización diferente para la matriz de entrada/salida, usando métodos supervisados y no-supervisados, revisando el rendimiento de cada uno de ellos, para observar cual se ajusta más a la naturaleza de la base de datos. A su vez, se podrían modificar parámetros internos del algoritmo genético, como modificar la tasa de mutación o experimentar otro tipo de recombinación para datos de naturaleza real.

Puesto que se detectó que el tamaño de población es un ítem fundamental en el buen funcionamiento del algoritmo de optimización, se puede orientar un trabajo futuro, a la implementación de técnicas que manipulen tamaños de población dinámicas y ver su comportamiento en comparación con técnicas estáticas como la manejada en el presente trabajo.

BIBLIOGRAFÍA

- [1] Remaining useful life estimation - A review on the statistical data driven approaches. Xiao-Sheng Si, Wenbin Wang, Chang-Hua Hu, Dong-Hua Zhou. European Journal of Operational Research, 2011.
- [2] Damage propagation modeling form aircraft engine run-to-failure simulation. Saxena A, Goebel K, Simon D, Eklund N. In: Proceedings of 2008 international conference on prognostics and health management, 2008.
- [3] A probabilistic-based airframe integrity management model. Wang X, Rabiei M, Hurtado J, Modarres M, Hoffman P. Reliability Engineering & System Safety, 2009.
- [4] Electronic-prognostics- a case study using global positioning system (GPS). Brown D, Kalgren P, Byington C, Roemer M. Microelectronics Reliability, 2007.
- [5] Prognostics and life beyond 60 years for nuclear power plants. Bond L, Ramuhalli P, Tawfik M, Lybeck N. In: 2011 IEEE International conference on prognostics and health management, 2011.
- [6] Proactive management of materials degradation for nuclear power plant systems. Bond L, Taylor T, Doctor S, Hull A, Malik S. In: IEEE International conference on prognostics and health management, 2008.
- [7] Prognostics models in medicine. Methods of information in Medicine-Methodik der information in der Medizin, Abu-Hanna A, Lucas P, 2001.
- [8] Another look at measures of forecast accuracy. Hyndman R, Koehler A. International Journal of forecasting, 2006.
- [9] Remaining useful life estimation based on stochastic deterioration models: A comparative study. Khanh Le Son, Mitra Fouladirad, Anne Barros, Eric Levrat, Benoit lung. Reliability Engineering and System Safety, 2012.
- [10] Recurrent neural networks for remaining useful life estimation. Heimes F. In: International conference on prognostics and health management, 2008.
- [11] Predicting remaining useful life of rotating machinery based artificial neural network. Abd Kadir Mahamad, Sharifah Saon, Takashi Hiyama, 2010.
- [12] Experience Based Approach for Li-ion Batteries RUL Prediction. R Khelif , B Chebel-Morello, N Zerhouni, 2015.

- [13] Development and validation of bearing diagnostics and prognostics tools using hums conditions indicators. He D, Bechhofer E. In: 2008 IEEE aerospace conference, IEEE, 2008.
- [14] Remaining Useful Life Estimation on the Non-Homogenous Gamma with Noise Deterioration based on Gibbs Filtering: A case Study. Khang Le Son, Mitra Fouladirad, Anne Barros, 2012.
- [15] "Covariates and random effects in a gamma process model with application to degradation and failure,". J. Lawless and M. Crowder. Lifetime Data Analysis, vol. 10, no. 3, pp. 213–227, 2004.
- [16] Remaining useful life prediction for an adaptive skew-Wiener process model. Zeyi Huang, Zhengguo Xu , Xiaojie Ke, Wenhai Wang, Youxian Sun. Mechanical Systems and Signal Processing, 2017.
- [17] Multi-sensor information based remaining useful life prediction with anticipated performance, IEEE Trans. M. Wei, M. Chen, D. Zhou, 2013.
- [18] A Wiener-process-based degradation model with a recursive filter algorithm for remaining useful life estimation. Xiao-Sheng Si, Wenbin Wang , Chang-Hua Hu , Mao-Yin Chen , Dong-Hua Zhou. Mechanical Systems and Signal Processing, 2013.
- [19] Gaussian Processes for Machine Learning, The MIT Press. Rasmussen C.E & C.K.I Williams (2006), ISBN 026218253X.
- [20] Remaining Useful Life Prognosis of bearing based on Gauss Process Regression. Sheng Hong, Zheng Zhou, 2012.
- [21] An integrated Gaussian process regression for prediction of remaining useful life of slow speed bearings based on acoustic emission. S.A. Aye, P.S.Heyns. Mechanical Systems and Signal Processing, 2017.
- [22] Gaussian Process Regression for Fingerprinting based localization. Sudhir Kumar, Rajesh M. Hedge, Niki Trigoni. Ad Hoc Networks, 2016.
- [23] Transfer function and transient estimation by Gaussian process regression in the frequency domain. John Lataire, Tianshi Chen. Automatica, 2016.
- [24] Fast single image super-resolution using sparse Gaussian process regression. Haijun Wang, Xinbo Gao, Kaibing Zhang, Jie Li. Signal Processing, 2017.

- [25] Can Li-Ion batteries be the panacea for automotive applications?. A.Opitz, P. Badami, L Shen, K Vignarooban, A.M. Kannan. Renewable and Sustainable Energy Reviews, 2017.
- [26] Battery Data Set, Nasa Ames Prognostics Data Repository, B. Saha, K. Goebel. Moffett Field, CA, 2007. <http://ti.arc.nasa.gov/tech/dash/pcoe/prognostics-data-repository/>.
- [27] Li-ion dynamics and state of charge estimation. Mingheng Li. Renewable Energy, 2017.
- [28] Model-based condition monitoring for lithium-ion batteries. Taesic Kim, Yebin Wang, Huazhen Fang, Zafer Sahinoglu, Toshihiro Wada, Satoshi Hara, Wei Qiao. Journal of Power Sources, 2015.
- [29] OCV hysteresis in Li-ion batteries including two-phase transition materiales. M.A.Roscher, O.S. Bohlen, J. Vetter. Int J Electrochem, 2011.
- [30] The genetic algorithm: A robust method for stress inversion. Prithvi Thakur, Deepak C. Srivastava, Pravin K. Gupta. Journal of Structural Geology, 2017.
- [31] Genetic algorithm used in simulation model: Application in a maintenance process. Khebbache-Hadji, S* Hani Y*, Lahiani N*, and El Mhamedi A, 2012.
- [32] Genetic Algorithm-based optimization for cutting parameters in turning process. Doriana M. D'Addonna, Roberto Teti, 2013.
- [33] An effective approach to estimating the parameters of software reliability growth using a real-valued genetic algorithm. Taeyoun Kim, Kwangkyu Lee, Jongmoon Baik. The Journal of Systems and Software, 2015.
- [34] A study of the applicability of modified genetic algorithms for the parameter estimation of software reliability modeling. Chao-Jung Hsu, Chin-Yu Huang. Proceedings of the 34th Annual IEEE International Computer Software and Applications Conference, 2010.
- [35] Distribution network reconfiguration using a genetic algorithm with varying population size. Morad Abdelaziz. Electric Power Systems Research, 2017.
- [36] Roulette-wheel selection via stochastic acceptance. Adam Lipowski, Dorota Lipowska. Physica A, 2012.
- [37] Genetic Algorithms in Search, Optimization, and Machine Learning. D.E. Goldberg, Addison-Wesley, Reading, Massachusetts, 1989.

- [38] An introduction to genetic algorithms. Mitchell Melanie, A bradford book the MIT Press, 1998.
- [39] A genetic algorithm for function optimization: a Matlab implementation NCSU-IE TR 95. Houck, C.R. Joines, J.A. Kay. M.G, 1995.
- [40] Applying adaptive algorithms to epistatic domains. L. Davis. Proceedings of the International Joint Conference on Artificial Intelligence., 1985.
- [41] A study of control parameters affecting online performance of genetic algorithms for function optimization. J.D. Schaffer, R.A. Caruna, L.J. Eshelman, R. Das. Proceedings of the Third International Conference on Genetic Algorithms, 1989.
- [42] Lithium-ion battery state of health estimation using ah-v characterization. Le D, Tang XD. In: Annual Conference of the prognostics and health management society; 2011. p. 367-73.
- [43] A comparative analysis of techniques for electric vehicle battery Prognostics and Health Management (PHM). Mohammad Rezvani , Mohamed AbuAli, Seungchul Lee, Jay Lee, Jun Ni, 2011.
- [44] Data Driven method based on particle swarm optimization and k-nearest neighbor regression for estimating capacity of Li-ion battery. Chao Hu, Gaurav Jain, Puqiang Zhang, Craig Schmidt, Parthasarathy Gomadam, Tom Gorka, Applied Energy , 2014.
- [45] Li-ion battery capacity estimation: A geometrical approach. Chen Lu, Laifa Tao, Huanzhen Fan. Journal of Power Sources, 2014.
- [46] A novel multistage Support Vector Machine based approach for Li ion battery remaining Useful Life estimation. Meru A. Patil, Piyush Tagade, Krishman S. Hariharan, Subramanya M. Kolake, Taewon Song, Taejung Yeo, Seokgwang Doo. Applied Energy , 2015.
- [47] Temperature Dependent power capability estimation of lithium-ion batteries for hybrid electric vehicles. Fangdan Zheng, Jiuchun Jiang, BingXiang Sun, Weige Zhang, Energy, 2016.
- [48] Distributed prognostic health management with gaussian process regression. S. Saha, B. Saha, A. Saxena, and K. Goebel,. IEEE Aerospace Conference, 2010.
- [49] Data driven prognostics for Lithium-ion Battery based on Gaussian Process Regression. Datong Liu, Jingyue Pang, Jianbao Zhou, Yu Peng, 2012.

- [50] A new prognostics method for state of health estimation of lithium-ion batteries based on a mixture of Gaussian process models and particle filter. Fan Li, Jiuping Xu. Microelectronics reliability, 2015.
- [51] A similarity-based prognostics approach for remaining useful life estimation of engineered systems. Wang T, Yu J, Siegel D, Lee J. In: International conference on prognostics and health management, 2008, pág. 1-6.
- [52] Modelo de un algoritmo genético con selección discriminatoria de individuos bajo un esquema de ponderación de probabilidades de mutación. Victor Manuel Arroyo Apaza. Universidad Católica de San Pablo, 2013.
- [53] Modelo para la valoración y predicción cuantitativa de la cicatrización empleando procesos Gaussianos de regresión. César Isaza, Julio Mosquera, 2009.
- [54] Modelo de maduración de fruto de banano empleando procesos Gaussianos de regresión y filtros acusto-ópticos. César Isaza, Julio Mosquera, 2009.
- [55] Detección de eventos sonoros en señales de música usando procesos Gaussianos. Pablo A. Alvarado-Durán, Mauricio A. Álvarez López, Álvaro A. Orozco-Gutierrez, 2013.
- [56] Análisis de expansión de redes de telefonía móvil basándose en indicadores claves de desempeño, utilizando procesos Gaussianos. Jhouben Janyk Cuesta Ramirez, 2013.
- [57] UN Anita permite calcular la vida remanente de los transformadores. Metacampus, Universidad Nacional de Colombia, 2017.
- [58] On-line remaining useful life prediction of Lithium-ion batteries based on the optimized gray model GM(1,1). Dong Zhou, Long Xue, Yijia Song, Jiayu Chen. Batteries, 2017.
- [59] State-of-Charge Estimation for Lithium-Ion Batteries Using Neural Networks and EKF. Charkhgard, M, Farrokhi, M. IEEE Trans. Ind. Electron. 2010.